



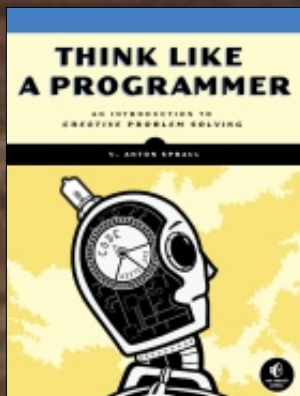
Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #67 - November 2012



Photo: Infidelic (Flickr.com)



BOOK REVIEW:
Think Like A
Programmer

STAY SECURE! CREATE A THIEF-PROOF COMPUTER

Full Circle Magazine is neither affiliated with, nor endorsed by, Canonical Ltd.



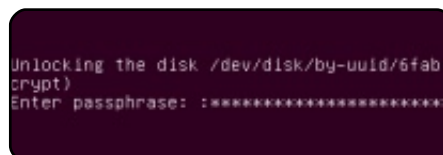
HowTo



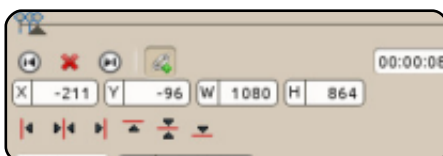
Programming Python 38 p.08

BookID	Integer [INTEGER]	The ID
Title	Text [VARCHAR]	The b
Published	Text [VARCHAR]	The Ye

LibreOffice - Part 19 p.09



Thief-Proof Computer p.12



Kdenlive - Part 4 p.26



Inkscape - Part 7 p.28



Graphics



Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

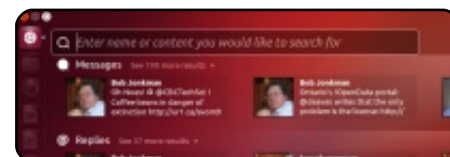
Columns

```
#An alias to make the
command more detailed
alias ls = 'ls -la --
color=always --classi
```

Command & Conquer p.06



Ask The New Guy p.35



Linux Labs p.39

BACK NEXT MONTH

Ubuntu Women p.XX



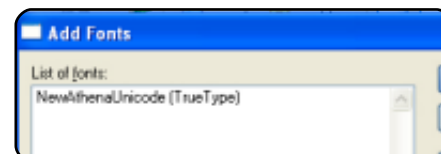
Ubuntu News p.04

BACK NEXT MONTH

Ubuntu Games p.XX

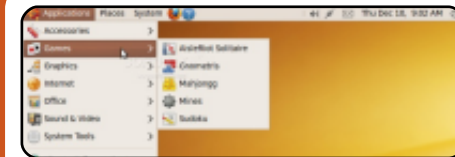


Q&A p.62



Closing Windows p.41

Opinions



My Story p.44



What Is... p.47



Review p.55



Letters p.60



Web Dev p.31



@ Web Dev



The articles contained in this magazine are released under the Creative Commons Attribution-Share Alike 3.0 Unported license. This means you can adapt, copy, distribute and transmit the articles but only under the following conditions: you must attribute the work to the original author in some way (at least a name, email or URL) and to this magazine by name ('Full Circle Magazine') and the URL www.fullcirclemagazine.org (but not attribute the article(s) in any way that suggests that they endorse you or your use of the work). If you alter, transform, or build upon this work, you must distribute the resulting work under the same, similar or a compatible license.

Full Circle magazine is entirely independent of Canonical, the sponsor of the Ubuntu projects, and the views and opinions in the magazine should in no way be assumed to have Canonical endorsement.



Welcome to another issue of Full Circle!

This month we have the return of Programming in Python, LibreOffice continues, and our third HowTo is an epic article on how to make your computer thief-proof. It's a long article, but I'm sure you'll find it interesting. The procedure is a bit fiddly, but, if the contents of your computer are valuable, then I'm pretty sure the long procedure will be well worth it in the end.

The graphics topics continue with Inkscape and the last of my Kdenlive articles. Next month, we'll begin a series on Blender. Yes, we've finally got a series on Blender. It's something I'm often emailed about; several people have promised articles (then not delivered), but, yes, a Blender series will begin next month.

Another book review for you this month. '**Think Like a Programmer**' from No Starch Press is the subject, and it's given a rave review by Lucas. Please think about buying a book from **No Starch Press** (<http://nostarch.com/>), they're a small publisher who help us by sending review copies when they can.

Anyway, enough rambling from me. Enjoy the issue, and we'll see you again next month for our last issue (of 2012).

All the best, and keep in touch!

Ronnie

ronnie@fullcirclemagazine.org



This magazine was created using :



Full Circle Podcast

Released monthly, each episode covers all the latest Ubuntu news, opinions, reviews, interviews and listener feedback. The Side-Pod is a new addition, it's an extra (irregular) short-form podcast which is intended to be a branch of the main podcast. It's somewhere to put all the general technology and non-Ubuntu stuff that doesn't fit in the main podcast.

Hosts:

- Les Pounder
- Tony Hughes
- Jon Chamberlain
- Oliver Clark



<http://fullcirclemagazine.org>



AUDIO MP3



AUDIO OGG

Download

Since the last issue of Full Circle Magazine the following has taken place:

Ubuntu Core on Nexus7

Jono Bacon writes about plans for the Ubuntu core on Nexus 7 project. Bacon pointed out very clearly: *"This is not going to be a tablet Unity interface running on the 8/16GB Nexus 7, but instead will focus on getting the current Ubuntu Desktop running on the Nexus so that we can ensure pieces such as the kernel, power management and other related areas are working effectively on a tablet device."*

Along with the project description, Bacon provides information for potential project contributors (testers and developers) and points out the UDS-R session schedule for getting Ubuntu core on Nexus 7.

<http://www.jonobacon.org/2012/10/26/ubuntu-core-on-the-nexus-7/>

Ubuntu 11.04 (Natty Narwhal) end-of-life reached on October 28, 2012

An email to the ubuntu-announce mailing list confirms that the support period for Ubuntu 11.04 (Natty Narwhal) formally ended on October 28, 2012 and Ubuntu Security Notices will no longer include information or updated packages for Ubuntu 11.04.

<https://lists.ubuntu.com/archives/ubuntu-announce/2012-October/000165.html>

Raring Ubuntu Developer Summit Complete!

During the week of 29 Oct - 1 Nov 2012 UDS-R took place with an overwhelming amount of news coming from those planning sessions for the Raring Ringtail 13.04 scheduled to be released next May.

Interviews and videos from UDS-R can be found at:
<http://www.youtube.com/user/ubuntudevelopers>

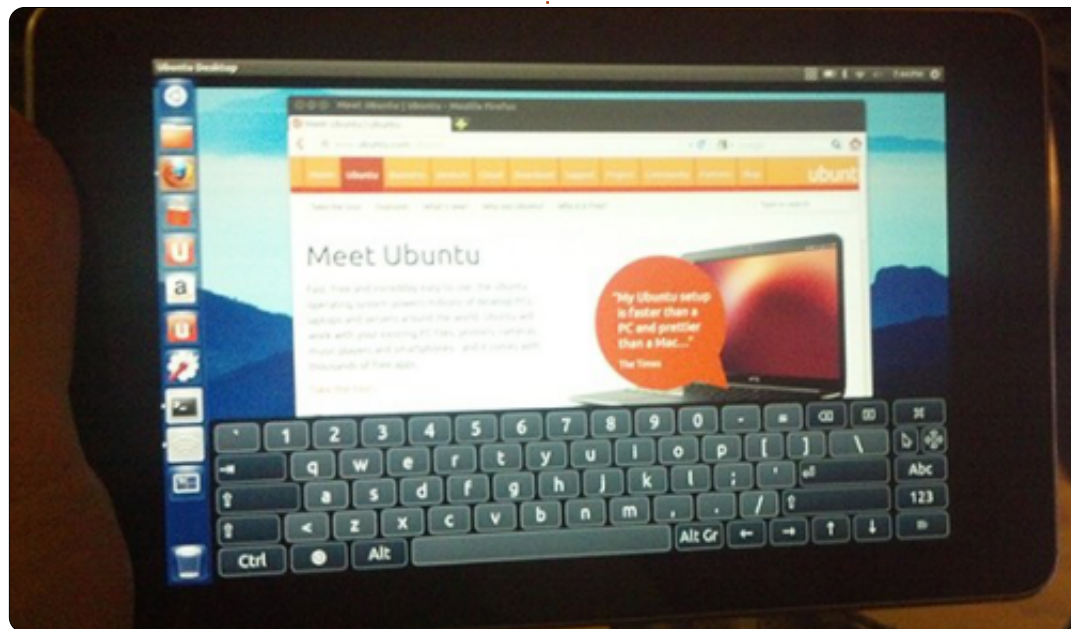
Steam for Linux Beta Now Available

Valve announces the launch of a limited access beta for its new Steam for Linux client, available for Ubuntu 12.04 users. It will be available for a widening group of users, including other distributions as soon it has reached a satisfactory level of stability.

<http://store.steampowered.com/news/9289/>

Mark Shuttleworth: Designing the Future, Together

Linux.com editorial staff brings us highlights of Mark Shuttleworth's keynote which he held at LinuxCon Europe 2012 in Barcelona. Those highlights include cloud and mobile driving



change, Ubuntu on every cloud, crowd-sourcing solutions, and form factors converging. Shuttleworth points out that "as we head toward the future of computing, we must focus on collaboration, communication, and integration at the operational level, not just the tech level."

<http://www.linux.com/news/enterprise/cloud-computing/661497-mark-shuttleworth-designing-the-future-together/>

Unity Technologies releases 4.0 with game export for Ubuntu

"The Unity cross-platform development tool version 4 has been released from beta into the growing world of Ubuntu game development." It brings with it many new features, including a Linux export feature that allows developers to easily publish their games to the Ubuntu Software Center.

<http://blog.canonical.com/2012/11/15/unity-technologies-releases-4-0-including-game-export-for-ubuntu/>

Ubuntu Core Desktop on the Nexus 7: Getting Involved

Jono Bacon writes that the efforts to getting the core components in Ubuntu working and optimized for the Nexus 7 tablet are underway, and there are many ways those interested in helping out can get involved. Developers can help by fixing bugs, optimizing software, and other tasks that benefit the Ubuntu Core Desktop on the tablet. Testers are also needed to run benchmarks and report bugs.

<http://www.jonobacon.org/2012/11/13/ubuntu-core-desktop-on-the-nexus-7-getting-involved/>

New Contributor Feedback – 12.10

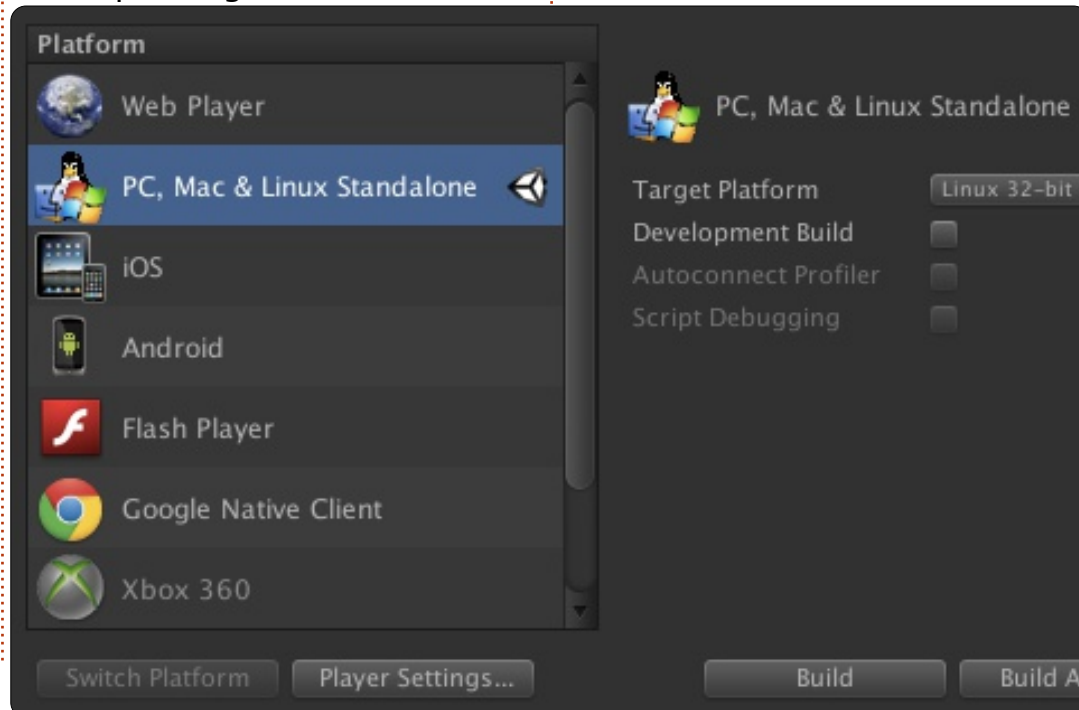
Vibhav Pant shares results from the New Contributor Feedback in the 12.10 cycle. *"We have summarized this feedback in the attached report. It is our hope that it will help drive further discussion about our development processes, tools, and documentation in the lead up to UDS and over the course*

of the next cycle."

<http://fridge.ubuntu.com/2012/11/19/new-contributor-feedback-12-10/>

Many Thanks to the Ubuntu News Team for their contribution this month.

News this month comes from:
Ubuntu Weekly Newsletter Issue 289 - <https://wiki.ubuntu.com/UbuntuWeeklyNewsletter/Issue289>
Ubuntu Weekly Newsletter Issue 290 - <https://wiki.ubuntu.com/UbuntuWeeklyNewsletter/Issue290>
Ubuntu Weekly Newsletter Issue 291 - <https://wiki.ubuntu.com/UbuntuWeeklyNewsletter/Issue291>
Ubuntu Weekly Newsletter Issue 292 - <https://wiki.ubuntu.com/UbuntuWeeklyNewsletter/Issue292>





This past month saw me installing Windows 8 on my dual monitor PC setup, in order to test it thoroughly for a client. Naturally, I didn't remove ArchLinux, but there are certain steps required to keep both operating systems happy. As such, I felt this month it may be useful to explain some of the extra steps, in order to save some people headaches when dealing with this.

Trick 1: Hard disk order

Note: This is important only to people with multiple physical hard disks in their PC.

Windows has a major complaint any time the primary hard disk is not the one Windows is installed in. It works just fine, so long as you don't do anything with the partition layout. In order to run an upgrade, for example, you need to have the main partition (Windows C: drive) set to active. This can be done in the disk management menu, but won't work if you've made the Windows drive

secondary. Solving this is as easy as switching the order in your BIOS.

Trick 2: GRUB is gone

As one would expect, Windows replaces any boot manager with its own. After this happens, you have two options: either create a menu for Linux from the Windows boot manager (this is possible), or re-install GRUB. If, like above, you have two disks, my preference is to simply install GRUB on the Linux drive, and leave the windows boot manager intact on the other. The boot manager used is always the one on the primary drive, so you can effectively switch between both managers by switching the hard disk order. This saves you the trouble. If, however, you don't have two drives, you'll need to re-install GRUB. The easiest way to do this is to boot from a live CD, and, for GRUB2 (which is used on all the newest Ubuntu versions), run:

```
sudo grub-install /dev/sdX
```

in the terminal. Remember to replace sdX with the actual hard

disk number (for example, /dev/sda). You can check this with either gparted, fdisk -l or any similar program. If you have internet in the live CD, you can also install a graphical tool called boot-repair.

Trick 3: System time is constantly wrong in both Windows and Linux

This is often the case when Ubuntu is set to use UTC time. Coordinated Universal Time (UTC) is a modern successor to GMT, and is the de facto standard for the Network Time Protocol and quite a bit of the internet. The problem? Windows refuses to offer an easy option to enable UTC. You can edit the registry in order to enable it (see UbuntuTime link in Further Reading below). The, probably easier, option is to set Linux to local time. To do so, simply do the following:

```
sudo vim /etc/default/rcS
```

Then edit, or create, the following line:

```
UTC=no
```

If you're someone who isn't comfortable editing files like this, or editing the registry, you can also simply adjust your timezone in Windows to balance the offset. For example, if you live in GMT+1, and your clock is 2 hours behind, switch to GMT+3, and it should work just fine. However, this may create unforeseen problems further down the line.

The "Developer-recommended" solution is to set both Windows and Linux to UTC, and disable Internet synchronization in Windows (it doesn't seem to work with UTC), instead relying upon NTP in Linux. I'm currently testing this solution, and it seems to be working alright so far.

Trick 4: Keeping media organized

If you want to keep your media

(music, videos, pictures, etc.) available to both systems without having to constantly copy and update the files, the easiest solution is to have a media partition. I've split my 1TB hard disk into 200GB for Linux, and 700GB for media. The Linux stuff is just a / and /home (both EXT4), and the 700GB partition is NTFS. The 700GB partition contains all my music, videos, pictures, and backups. Once you've relocated the files onto the media partition, you need to create symbolic links to them in Linux. My preference is to create a sub-folder link, as opposed to replacing the Music, Pictures and Videos folder completely. To create a symbolic link, do the following:

```
ln -s /media/Media/Music  
~/Music/External
```

This would create a link to your Music folder (on the partition called "Media"). The link would be called External, and is located in the Music folder within your home folder. You can, naturally, change the command however you see fit. You can do something similar in Windows 7 or 8 (possibly in Vista), by adding a new folder to your "Libraries" for Music, Pictures,

Videos. Instructions can be found in the Further Reading section.

A quick note: If Ubuntu doesn't automatically connect the Media partition, you'll need to do that manually, and instructions are below in the Further Reading section.

This trick for media files can also be used for any other type of file you would need to share. For example, you can stick your Dropbox folder on it, and avoid having to download any updates twice.

I hope at least a few of my readers find this article useful, and for those of you who didn't, odds are I'll have something to interest you next month. If you have any questions, comments, or suggestions, feel free to email me at lswest34@gmail.com. This month I've also written a review on Think Like A Programmer by V. Anton Spraul, so if you're interested in programming, you may want to give it a look.

Further Reading:

<http://windows.microsoft.com/is-IS/windows7/Customize-a-library> – Customize Windows Library

<https://help.ubuntu.com/community/UbuntuTime> – UbuntuTime

<https://help.ubuntu.com/community/RecoveringUbuntuAfterInstallingWindows> – Restore GRUB

<https://help.ubuntu.com/community/AutomaticallyMountPartitions> – Automatically Mount Partitions

<http://superuser.com/questions/482860/does-windows-8-support-utc-as-bios-time> - UTC in Windows



Lucas has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. You can email Lucas at: lswest34@gmail.com.

Full Circle Podcast Episode 31, The Difficult Third Episode!!

NEW

It may be a new team of podcasters, but the format is still the same.



Your hosts:

- Les Pounder
- Tony Hughes
- Jon Chamberlain
- Oliver Clark

All members of the Blackpool (UK) LUG
<http://blackpool.lug.org.uk>

Olly & Tony bring everyone up to date with what's been going on in the "Summer Break", Olly talks about his Gnomebuntu install, Code-academy and Tony talks about his new Nexus 7.

Download



HOW-TO

Written by Greg Walters

Programming In Python: Pt 38

As I promised in part 37, we will take the transposer app that we created, and create an APK to install it on your android device.

Before we get started, let's make sure we have everything ready. First thing we need is the two files we created last time in a folder that you can easily access. Let's call it "transposer". Create it in your home directory. Next, copy the two files (transpose.kv and transpose.py) into that folder. Now rename transpose.py to main.py. This part is important.

Next, we need to reference the kivy packaging instructions in a web browser. The link is <http://kivy.org/docs/guide/packaging-android.html>. We will be using this for the next steps, but not exactly as the Kivy people intended. You should have the android SDK from our earlier lesson. Ideally, you will go through and get all the software that is listed there, but for our purposes, you can just follow along here. You

```
./build.py --dir <path to your app>
--name "<title>"
--package <org.of.your.app>
--version <human version>
--icon <path to an icon to use>
--orientation <landscape|portrait>
--permission <android permission like VIBRATE> (multiple allowed)
<debug|release> <install|installr|...>
```

will need to download the python-for-android software. Open a terminal window and type the following...

```
git clone
git://github.com/kivy/python-
for-android
```

This will download and set up the software that we need to continue. Now, in a terminal window, change your directory to the python-for-android/dist/default folder.

Now you will find a file called build.py. This is what will do all the work for us. Now comes the magic.

The build.py program will take various command-line arguments and create the APK for you. Shown above is the syntax for build.py

taken directly from the Kivy documentation.

For our use, we will use the following command (the "\" is a line continuation character):

```
./build.py --dir ~/transposer
--package
org.RainyDay.transposer \
--name "RainyDay Transposer"
--version 1.0.0 debug
```

Let's look at the pieces of the command...

./build.py - this is the application
--dir ~/transposer - this is the directory where our application code lives.
--package org.RainyDay.transposer - This is the name of the package
--name "RainyDay Transposer" -

this is the name of the application that will show up in the apps drawer.

--version 1.0.0 - the version of our application
debug - this is the level of release (debug or release)

Once you execute this, assuming that everything worked as expected, you should have a number of files in the /bin folder. The one you are looking for is titled "RainyDayTransposer-1.0.0-debug.apk". You can copy this to your android device using your favorite file manager app, and install it just like any other application from the various app stores.

That's all I have time for this month.





HOW-TO

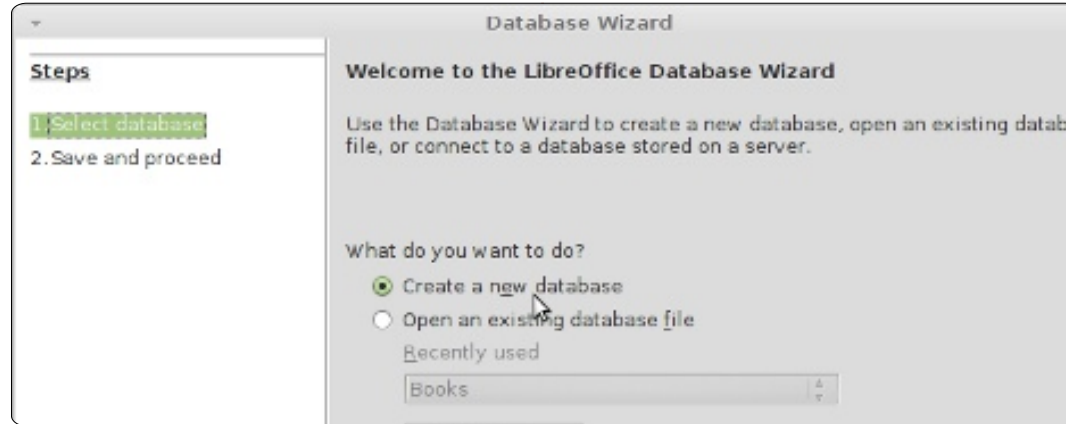
Written by Elmer Perry

LibreOffice Pt20: Base

Databases are used to store information about objects or data. In the previous tutorial, we mapped out how our books database would look. We designed tables for our data, and defined relationships between those tables. Now, we will put our planning into action by actually creating the database file, adding the tables, and creating the relationships.

Creating the Database File

As I mentioned before, Base is not a database file but an interface for accessing and manipulating a database file. Although it is



possible to connect to many different database types, we will use the default HSQL database for our books database.

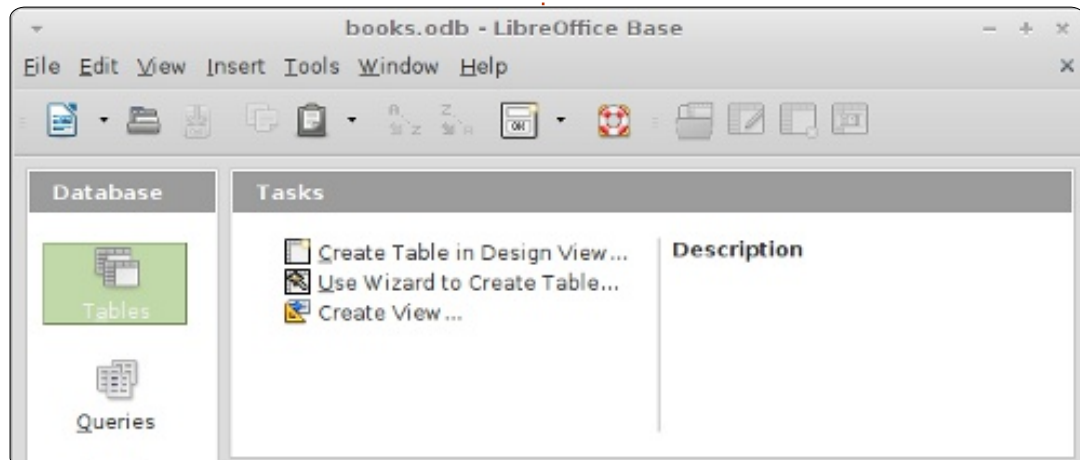
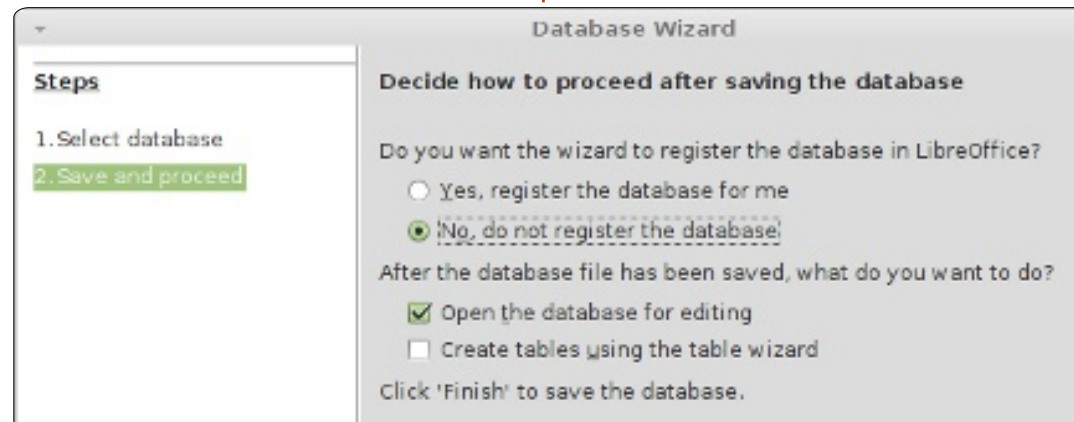
To start the database wizard, select Database from the LibreOffice Home screen or File > New > Database. The first screen of the database wizard lets us choose whether we want to open

an existing database or create a new one. Select Create a New Database, and click Next.

The second screen of the wizard asks us whether we want to register the database and what we want to do once the database has been created. Registering a database in LibreOffice makes it

available in all our documents. We won't need this for our database, so select No – do not register the database. Check Open the Database for Editing, and click Finish. LibreOffice will open a file dialog to define a location and name for the database. I simply named the file: books

Once you have a name and location for the database file, the main Base screen opens. Down the left side, you have the different pieces which can make up a database file. The top right gives you access to the different actions you can take for each part, and the lower right shows the objects already created.



Field Types

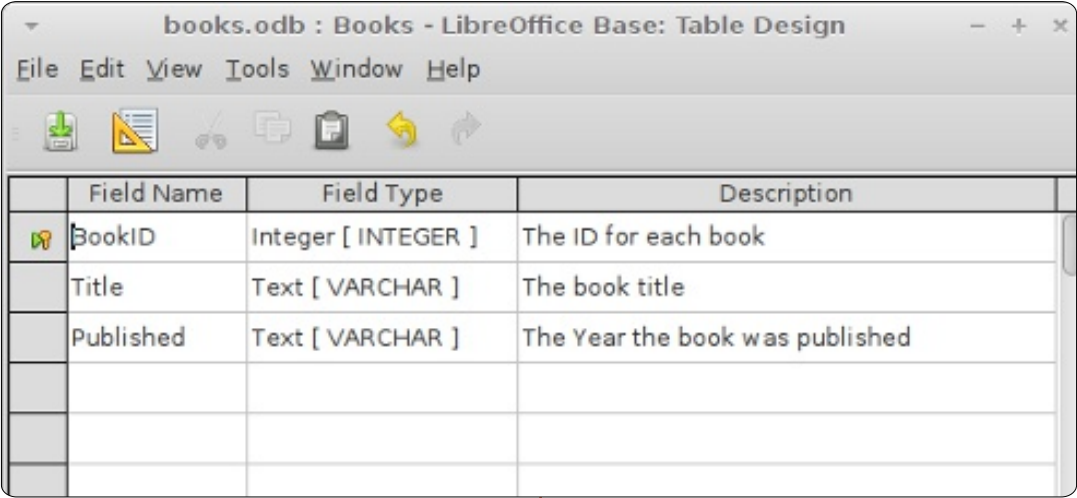
Before we create our first table, we need to discuss some of the common database field types. When you select a type for a field, you are presented with many options for the type. Many of the types are identical, and are there for compatibility reasons. The most common types are:


- Integer** – a whole number, eg. 123
- VarChar** – a variable length string of characters. You will define the maximum length for the VarChar.
- Date** – a date value, of course, eg. 10-15-2012 (the exact format is location specific)
- Time** – a time value, such as 09:15:25
- decimal** – a real number including the whole (integer) number and its fractional part, eg. 123.25 (the part separator is location specific)

For our purposes, we will use Integer and VarChar.

Creating the Tables

Base has three different ways to create tables: through a table wizard, through design view, and by SQL statements. The table



	Field Name	Field Type	Description
	BookID	Integer [INTEGER]	The ID for each book
	Title	Text [VARCHAR]	The book title
	Published	Text [VARCHAR]	The Year the book was published

wizard is good only to create specific types of tables by picking from a list of predefined field names. The SQL method requires you to know and understand the SQL language and is beyond the scope of this article. The design view is usually the best choice, and presents you with a list you fill in to create your table. We will use the design view to create our tables for this project.

We will start with the Books table. Select Tables from the Database pane on the left. In the Tasks pane, click on Create Table in Design View... to open the Design View dialog. Across the top you have labels for each of the elements of a field: Field Name, Field Type, and Description. The Description is optional but is

useful for making notes about how a field is used. At the bottom, we see the Field Properties. This section will change according to the type of field we select.

In the first field, enter the name BookID. From the dropdown box in Field Type, select Integer. Adding a description is up to you. Under the field properties, change AutoValue to Yes. This will place a key icon in the box beside the field record showing it is the primary (or key) index. In the second row, type Title for the name. Give this one a type VarChar. Again, a description is up to you. In the field properties, leave the length at 100, the default for VarChar. The third field is Published with a type of VarChar. Change the length in the field properties to 12. I chose VarChar

rather than date because we just want the year, and if the publishing date of a book is unknown, I can enter just "Unknown". Click on the save icon, and Base prompts you for a table name. Enter Books.

Our tables for Authors and Media are created in much the same way. For Authors, create two fields: AuthorID, integer (AutoValue Yes); and Name, VarChar (length 50). For Media, MediaID, integer (AutoValue Yes); and Type, VarChar (length 25).

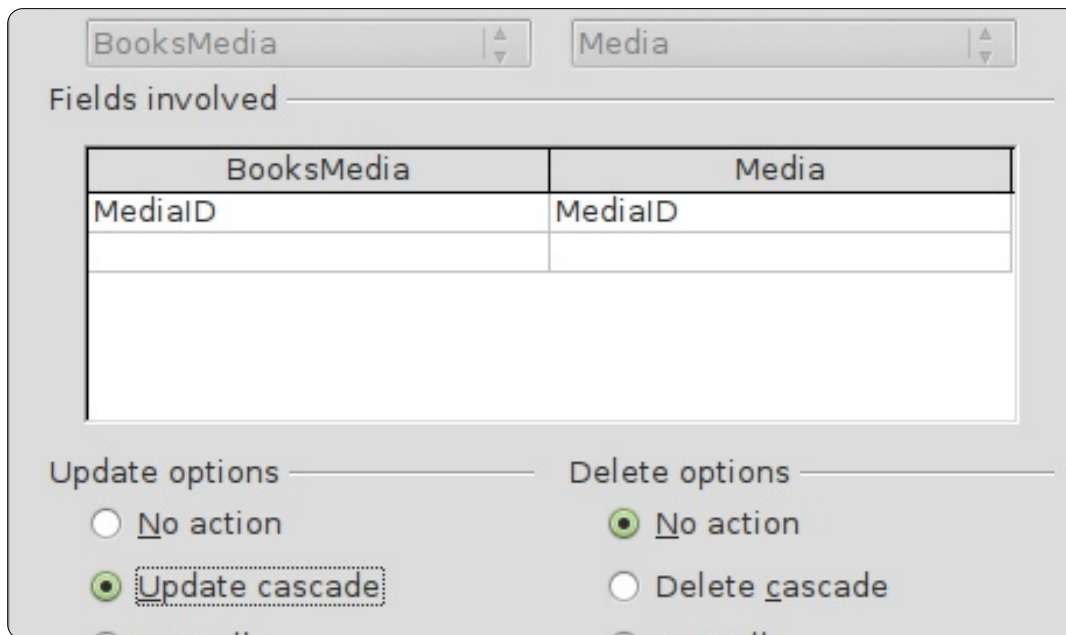
Our two foreign key tables need a little different treatment. In BooksAuthors, create two integer fields named BookID and AuthorID. Click on the icon box beside the first record. Holding down the Shift key, click in the icon box for the second. At this point, you should have both records selected. Right-click the icon box and select Primary Key from the context menu. This creates a combination key. The two values together create the primary key, which uniquely identifies each record in the table. For the BooksMedia table, create two integer fields named BookID and MediaID. Select both fields, right-click, and select Primary Key.

Create Relationships

Once we have all our tables defined, we can create the relationships that bind everything together. We will create relationships between our three main tables and our foreign key tables. The direction in which you drag the fields is important, so pay close attention to how you do it.

To start the Relation Design dialog, go to Tools > Relationships. You are presented with a list of tables. Select a table and click Add to add the table to the Relation Design. Add the tables in the following order to make it easy: Authors, BooksAuthors, Books, BooksMedia, Media. Once all the tables are added, select Close.

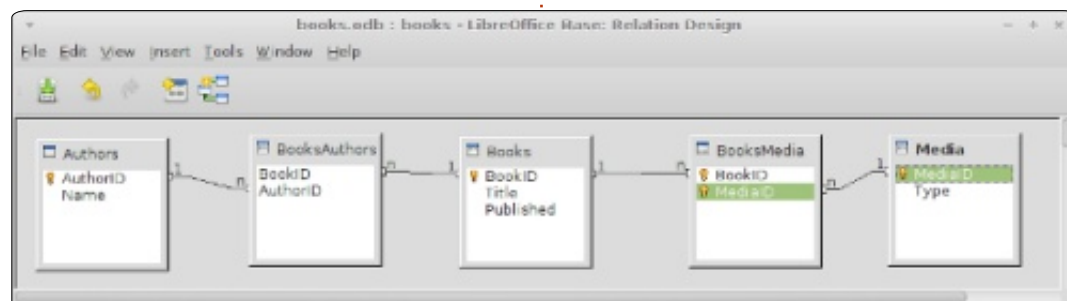
Drag the BookID field in Books to BookID in BooksAuthors. A Relation dialog pops up. Under Update option, pick Update cascade and OK. This will cause the



field to update when the Books table updates. Drag the AuthorID in Authors to AuthorID in BooksAuthors. Select Update cascade in the Relation dialog. Next, drag the BookID in Books to BookID in BooksMedia. Select Update cascade. Finally, drag MediaID in Media to MediaID in BooksMedia. Select Update cascade. Your relation design should look something like the one

pictured below.

With our tables and relationships created, we are ready to begin work on creating forms for data input. In our next How-To, we will create the forms for data entry. Everything will come together to create a usable data entry system.



The Ubuntu Podcast covers all the latest news and issues facing Ubuntu Linux users and Free Software fans in general. The show appeals to the newest user and the oldest coder. Our discussions cover the development of Ubuntu but aren't overly technical. We are lucky enough to have some great guests on the show, telling us first hand about the latest exciting developments they are working on, in a way that we can all understand! We also talk about the Ubuntu community and what it gets up to.

The show is presented by members of the UK's Ubuntu Linux community. Because it is covered by the Ubuntu Code of Conduct it is suitable for all.

The show is broadcast live every fortnight on a Tuesday evening (British time) and is available for download the following day.

podcast.ubuntu-uk.org



Elmer Perry's history of working, and programming, computers involves an Apple IIE, adding some Amiga, a generous helping of DOS and Windows, a dash of Unix, and blend well with Linux and Ubuntu.



HOW-TO

Written by Xavier Berger

Create A Thief-Proof Computer

In our modern society, computers become tools hosting a lot of private information. Losing these data or displaying these data to the public could have a big impact for the owner.

In this article we will see how to:

1. Prevent the computer from booting on the secured system without a startup usb key
The first barrier will be to prevent the PC from booting when it is not in the hands of its owner. To achieve this goal, we will "split" the hardware into two pieces. When these two pieces are joined, the computer can be used (and boot); if not, the computer will not start. As many people have to purchase Windows with a laptop, we will see how to give access to this OS when the key is not present. This could be useful if you want to lend your PC to a friend to give her access to the Internet.

2. Prevent the data from unwanted access

If a person can access my disk, he should still not have access to the data. The file system and swap contain or can contain personal data. We will see how to encrypt the data at the level of the partition, and keep our data safe.

3. Prevent data loss

To prevent data loss, doing regular backup is something mandatory, but, if the backup storage is located in the same building as the computer, it may be robbed/destroyed as well. To prevent data loss, we need to externalize the data. The cloud is a good solution for such an externalization. It could be done in real time, and doesn't require any discipline from the end user. To ensure the privacy of the data in such cases, we will also encrypt the data before synchronizing it to the cloud. The data in the cloud is a mirror of the current data in the computer.

4. Use the created usb key as a toolbox by adding live distributions

With the usb key in our hands, we

will see how to add an additional live distributions that could be useful for troubleshooting or rescue.

Installation

To implement the procedure described below, you will need the following items:

- A target computer (with or without an existing OS running)
- Two blank CD-ROMs – to burn Ubuntu installation media
- One USB key (min > 1GB, recommended > 4GB)
- A computer with an operating

system running (that could be the target)

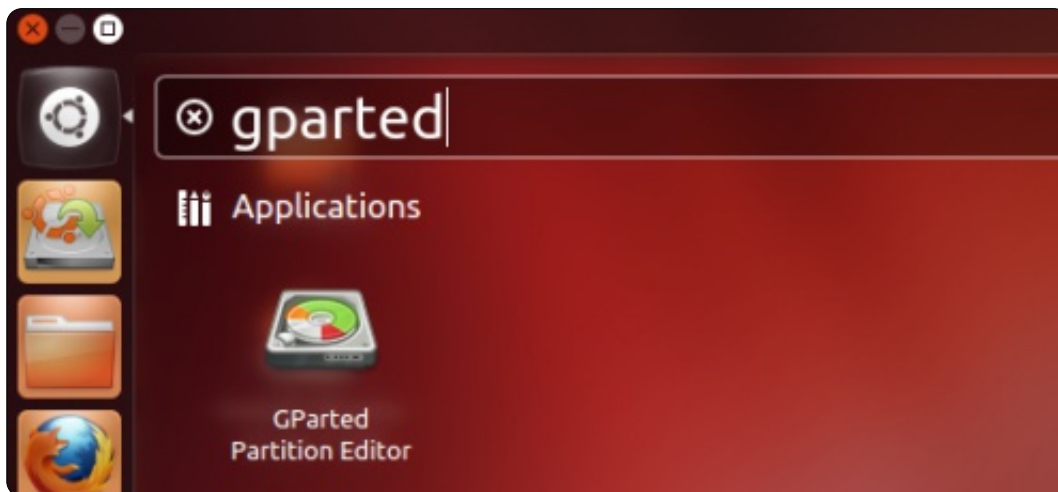
Preparation

Download *ubuntu-12.04.1-desktop-i386.iso* from <http://releases.ubuntu.com/precise/> and burn it to a CD-ROM.

Plug in the usb key, and boot from the Ubuntu 12.04.1 desktop CD-ROM freshly burnt.

In the welcome screen, select Try Ubuntu.





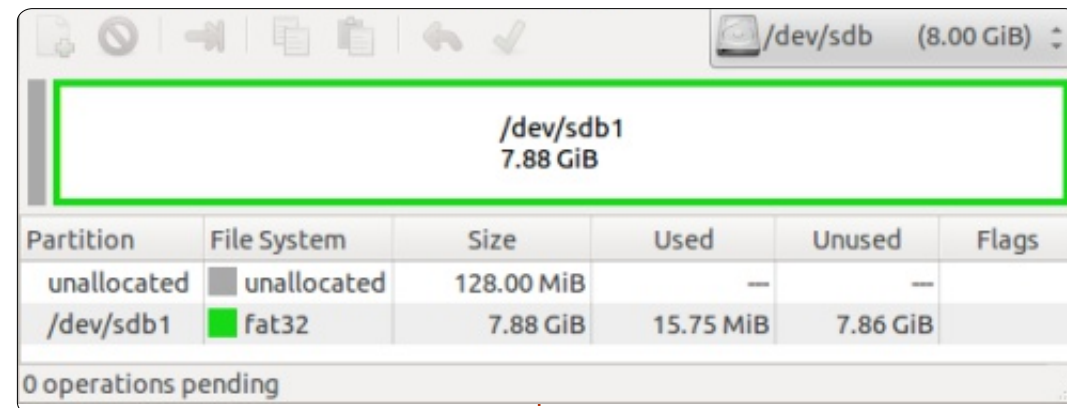
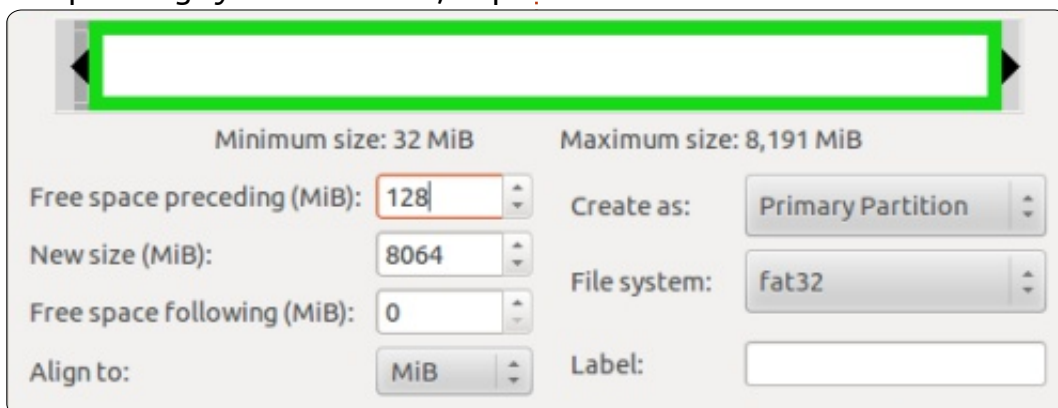
Once the live system has booted, start gparted to prepare the usb key.

Select the usb key (/dev/sdb), and create a FAT32 partition preceded by a free space of 128MB. This space will be used later by the secured operating system.

If your computer doesn't have an operating system installed, skip

the next step of this chapter and continue with the following section.

If you purchased Microsoft Windows with the laptop, we will keep it in the machine and available (in case you would like to lend your PC to your friends). First perform a backup of windows using the tools provided by your manufacturer... you may need or will have to restore Windows once

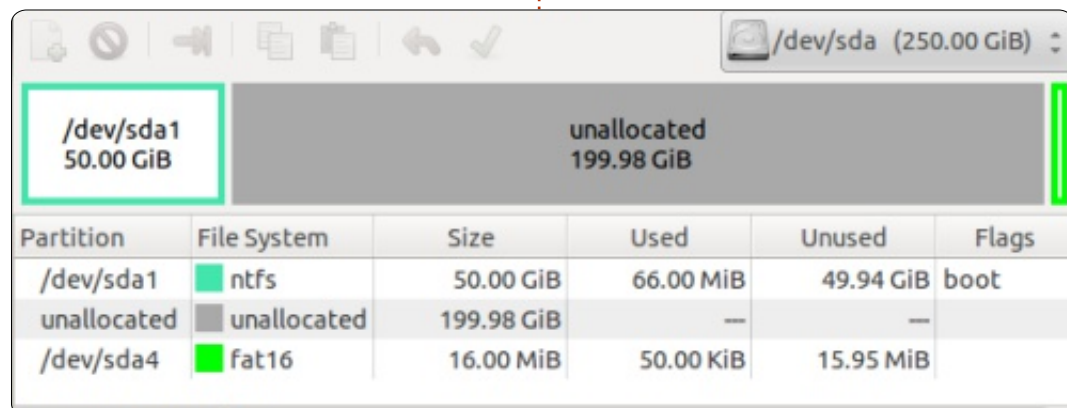


the partition resizing is done. Resize the existing windows partition to leave space for the real operating system.

- Boot on Ubuntu 12.04.1 desktop
- Start GParted
- Select the disk of the computer (/dev/sda)
- Resize the windows partition to 50GB, it should be enough for all occasions when you need to use this OS
- Remove unneeded partitions – to create free space that will be used to install Linux

Installation

During the installation, we will configure the Operating System to encrypt the data stored in the disk. This encryption will ensure the security of the data. The longer the encryption key, the better the protection, but this also lengthens the time to encrypt and decrypt. In this article, we choose the shortest proposed length for the key: AES 128-bit should be fast and secure enough. A key of 128-bits give



HOWTO - CREATE A THIEF-PROOF COMPUTER

about $3,4 \times 10^{38}$ possibilities.

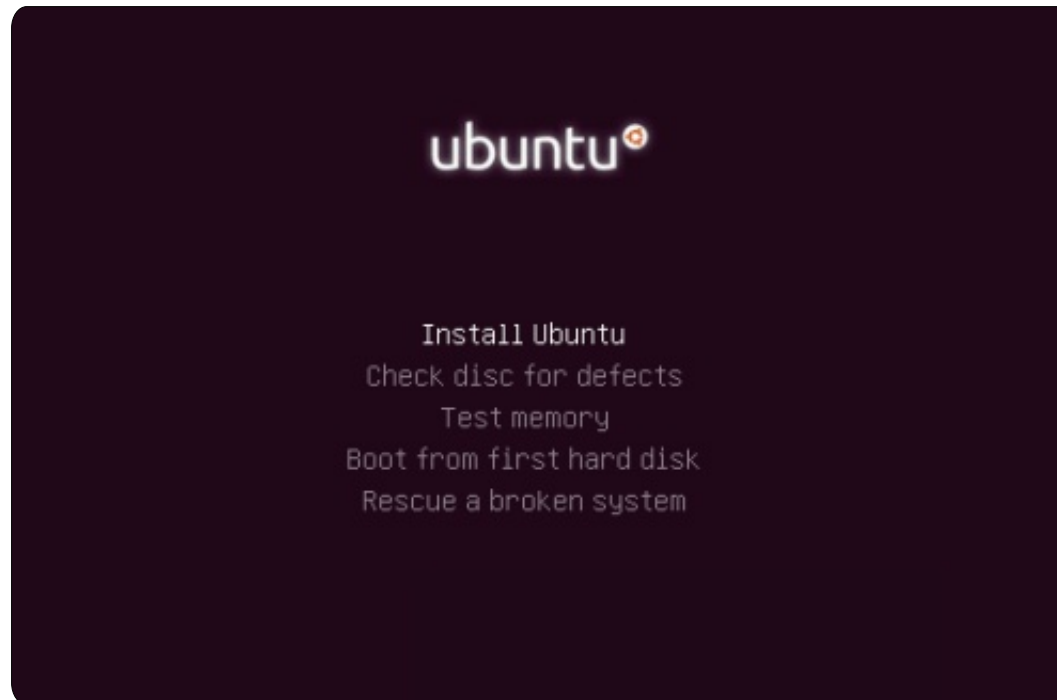
To understand how secure 128-bit keys are, you may read the analogy by Jon Callas at:

<http://www.interesting-people.org/archives/interesting-people/200607/msg00058.html>

"Imagine a computer that is the size of a grain of sand that can test keys against some encrypted data. Also imagine that it can test a key in the amount of time it takes light to cross it. Then consider a cluster of these computers, so many that if you covered the earth with them, they would cover the whole planet to the height of 1 meter. The cluster of computers would crack a 128-bit key on average in 1,000 years."

Even if you don't believe that the NSA has another planet devoted to key cracking, you still may want to use a longer key. If a weakness in your chosen crypto-module is found, it may limit the key space that needs to be tested, and you will then have an effectively shorter key. Using a 256-bit key will keep your data secure much longer if that should happen.

Download Ubuntu and burn it



```
overview of your currently configured partitions and mount points
to modify its settings (file system, mount point, etc.), a free
, or a device to initialize its partition table.
```

```
Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

SCSI1 (0,0,0) (sda) - 268.4 GB ATA VBOX HARDDISK
#1  primary   53.7 GB  B    ntfs
    pri/log   214.7 GB      FREE SPACE
#4  primary   16.8 MB      fat16
SCSI2 (0,0,0) (sdb) - 8.6 GB ATA VBOX HARDDISK
    pri/log   128.9 MB      FREE SPACE
#5  logical    8.5 GB

Undo changes to partitions
Finish partitioning and write changes to disk
```

on the second CD-ROM.

Boot on the freshly burnt CD-ROM. Select the language and Install Ubuntu.

Follow these instructions to install the system:

- Select the language to be used during the installation process: English
- Select your location, configure locales, configure the keyboard
- Define hostname, user and password
- Do not choose to encrypt the home directory. We will encrypt all the partition.
- Set clock and timezone
- To Partition disks, select Manual

Creating the /boot partition unencrypted

- Select the freespace on sdb and press enter
- Select Create a new partition

Define the size: keep the proposed size

- Type of the new partition: Primary
- Use As: Ext2 file system
- Mount point: /boot
- Bootable flag: on
- Select Done setting up the

HOWTO - CREATE A THIEF-PROOF COMPUTER

partition

Create a Logical partition

- Select the free space on sda and press enter
- Select Create a new partition
- Define the size: 128M

Type of the new partition: Primary

Location for the new partition:

Beginning

Use as: do not use

- Select Done setting up the partition

Create a Logical partition

- Select the free space on sda and press enter
- Select Create a new partition
- Define the size: keep the proposed size – which should be the maximum space available
- Type of the new partition: Logical
- Use as: do not use
- Select Done setting up the partition

Encrypt partition

- Select Configure encrypted volumes
- Write the change to disk and configure encrypted volumes: Yes
- Select Create Encrypted volumes
- Select: [*] /dev/sda5
- Key size: 128
- Done setting up the partition

```
Encrypted volume (sda5_crypt) - 214.6 GB Linux device-mapper (crypt)
#1          214.6 GB      f  ext4
SCSI1 (0,0,0) (sda) - 268.4 GB ATA VBOX HARDDISK
#1 primary   53.7 GB      B      ntfs
#2 primary   127.9 MB      ext2
#5 logical   214.6 GB      K  crypto  (sda5_crypt)
#4 primary   16.8 MB      fat16
SCSI2 (0,0,0) (sdb) - 8.6 GB ATA VBOX HARDDISK
#1 primary   126.9 MB      B  K  ext2    /boot
#5 logical    8.5 GB
```

- Keep current partition layout and configure encrypted volume: Yes
- Select Finish
- Enter a passphrase twice

Create LVM group and volumes

- Select Configuring the Logical Volume Manager
- Write change to disk and configure LVM : Yes
- Select Create volume group

- Volume group name: VolGroup
- Device for new volume group: [*] /dev/mapper/sda5_crypt
- Keep current partition layout and configure LVM: Yes
- Select Create logical volume
- Volume group: VolGroup
- Logical volume name: LV_slash
- logical volume size: 50GB
- Select Create logical volume
- Volume group: VolGroup

```
LVM VG VolGroup, LV LV_home - 162.6 GB Linux device-mapper (linear)
#1          162.6 GB      f  ext4    /home
LVM VG VolGroup, LV LV_slash - 50.0 GB Linux device-mapper (linear)
#1          50.0 GB      f  ext4    /
LVM VG VolGroup, LV LV_swap - 2.0 GB Linux device-mapper (linear)
#1          2.0 GB      f  swap     swap
Encrypted volume (sda5_crypt) - 214.6 GB Linux device-mapper (crypt)
#1          214.6 GB      K  lvm
SCSI1 (0,0,0) (sda) - 268.4 GB ATA VBOX HARDDISK
#1 primary   53.7 GB      B      ntfs
#2 primary   127.9 MB      ext2
#5 logical   214.6 GB      K  crypto  (sda5_crypt)
#4 primary   16.8 MB      fat16
SCSI2 (0,0,0) (sdb) - 8.6 GB ATA VBOX HARDDISK
#1 primary   126.9 MB      B  K  ext2    /boot
#5 logical    8.5 GB
```

Undo changes to partitions

Finish partitioning and write changes to disk

- Logical volume name: LV_swap
- logical volume size: 2GB
- Select Create logical volume
- Volume group: VolGroup
- Logical volume name: LV_home
- Logical volume size: keep the proposed size
- Select Finish

Configure the mounting points of LVM volumes

- Select LV_home --> #1
- Use as : Ext4 journaling file system
- Mount point: /home
- Options: [*] noatime (--> we don't wish to record the last time the file has been read)
- Select Done setting up the partition
- Select LV_slash --> #1
- Use as : Ext4 journaling file system
- Mount point: /
- Options: [*] noatime
- Select Done setting up the partition
- Select LV_swap --> #1
- Use as : swap area
- Select Done setting up the partition

Finish partitioning and write changes to disk

- Write changes to disks

HOWTO - CREATE A THIEF-PROOF COMPUTER

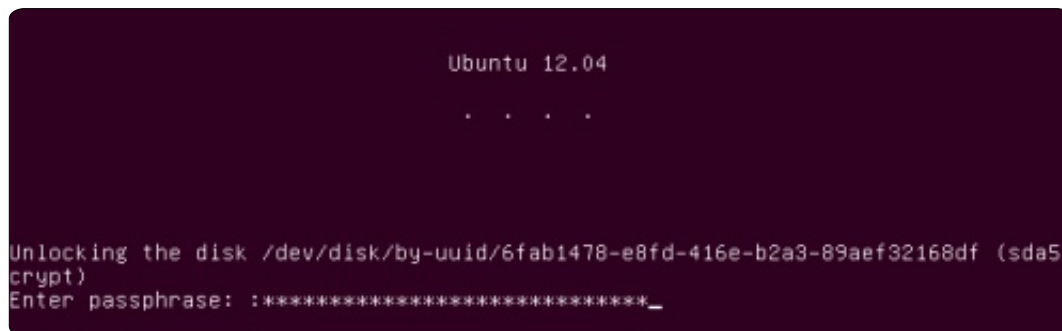
- Install GRUB on /dev/sdb (the key will then be required to boot on the secured system)
- Set the system clock to UTC: Yes
- Complete the installation

Congratulations, you have now a system where your data is encrypted and needs an external usb key to start. As we didn't touch the mbr of the hdd, the previous operating system should continue to start as it did previously. It is now required to explicitly boot with the usb key to access the secured area. When doing so, you will see the boot screen asking the password required to decrypt the disk.

We will see in the next chapter how to make our secured computer as easy to use as an unencrypted and unsecured computer, and we will configure it to ensure the sustainability of our data.

Configuration

In this section we will see how to customize the system to improve its usability and to protect our data from loss.



Warning: Unless specified, the commands below must be executed as root.

Using a label for the USB key will allow us to duplicate this key and ensure that the system will recognize it as the booting device.

Define the label for /dev/sdb2 as BOOT (/dev/sdb2 is the /boot partition)

```
e2label /dev/sdb2 BOOT
```

Update /etc/fstab to use label instead of uuid

```
replace UUID=(...) by  
LABEL=BOOT
```

Edit /etc/default/grub, uncomment to not use uuid in grub, and regenerate grub.cfg:

```
uncomment  
GRUB_DISABLE_LINUX_UUID=true
```

Insert the blue text (shown below) into the file /usr/lib/grub/grub-mkconfig_lib to configure grub to use label (if available in a volume)

Upgrade grub with the following command:

```
update-grub
```

We will now configure the system to decrypt the partition based on a file stored in the

startup key. The computer will then recognize the startup key and decrypt the partition without asking for a password anymore.

Create a keyfile in /boot

```
dd if=/dev/urandom  
of=/boot/keyfile bs=512  
count=4
```

```
chmod 444 /boot/keyfile
```

Add the new key into it as a valid key to decrypt the disk:

```
cryptsetup luksAddKey  
/dev/sda5 keyfile
```

Note: The contents of the file is important, not the filename.

Modify /etc/crypttab to use the boot key. Replace:

```
sdb5_crypt UUID=(...) none  
luks
```

with

```
if label="\${grub_probe} --device ${device} --target=fs_label 2> /dev/null`" ; then  
    echo "search --no-floppy --label ${label} --set root"  
elif fs_uuid="\${grub_probe} --device ${device} --target=fs_uuid 2> /dev/null`" ; then  
    echo "search --no-floppy --fs-uuid --set ${fs_uuid}"  
fi
```

HOWTO - CREATE A THIEF-PROOF COMPUTER

```
sdb5_crypt UUID=(...)  
/dev/disk/by-  
label/BOOT:/keyfile  
luks,keysript=/lib/cryptsetu  
p/scripts/passdev
```

Finally, update the initramfs with:

```
update-initramfs -uv
```

To remove auto decryption and reactivate passphrase only, modify /etc/crypttab by adding the boot key and remove the line added. The file should then look something like this

```
sdb5_crypt UUID=(...) none  
luks
```

Finally, update the initramfs:

```
update-initramfs -uv
```

Booting from the main disk instead of startup key

Some BIOS don't really like to boot from an external usb drive. In such a situation, we will create a screen displayed at boot that will allow the user to select which operating system to startup. We will use BURG which has a nice-

looking graphical interface. This solution will modify the MBR of the HDD, so then you have to pay particular attention to the action proposed here.

First create symlinks vmlinuz and initrd to the latest kernel and initrd image:

```
cd /boot  
  
ln -s vmlinuz-... vmlinuz  
  
ln -s initrd.img-...  
initrd.img
```

Warning: After every kernel upgrade, you will have to update these links pointing to the latest kernel

Mount the /boot partition of HDD:

```
umount /boot  
  
mount /dev/sda2 /boot
```

Install BURG on the system:

```
apt-get install python-  
software-properties  
  
add-apt-repository ppa:n-  
muench/burg  
  
apt-get update
```

```
apt-get install burg
```

Configure BURG to run from the internal HDD:

- Accept default parameters
- Select /dev/sda

Edit /etc/burg/30_osprober and append the keyword exit at the beginning of the file.

Edit /etc/burg.d/10_linux and append the keyword exit before the last while.

Update custom menu to boot on Linux or Windows:

```
menuentry "Linux" --class  
ubuntu {
```

```
insmod ext2  
set root=(hd1,2)  
linux /vmlinuz quiet splash  
initrd /initrd.img  
}  
menuentry "Windows" --class  
windows {  
    set root=(hd0,1)  
    chainloader +1  
}
```

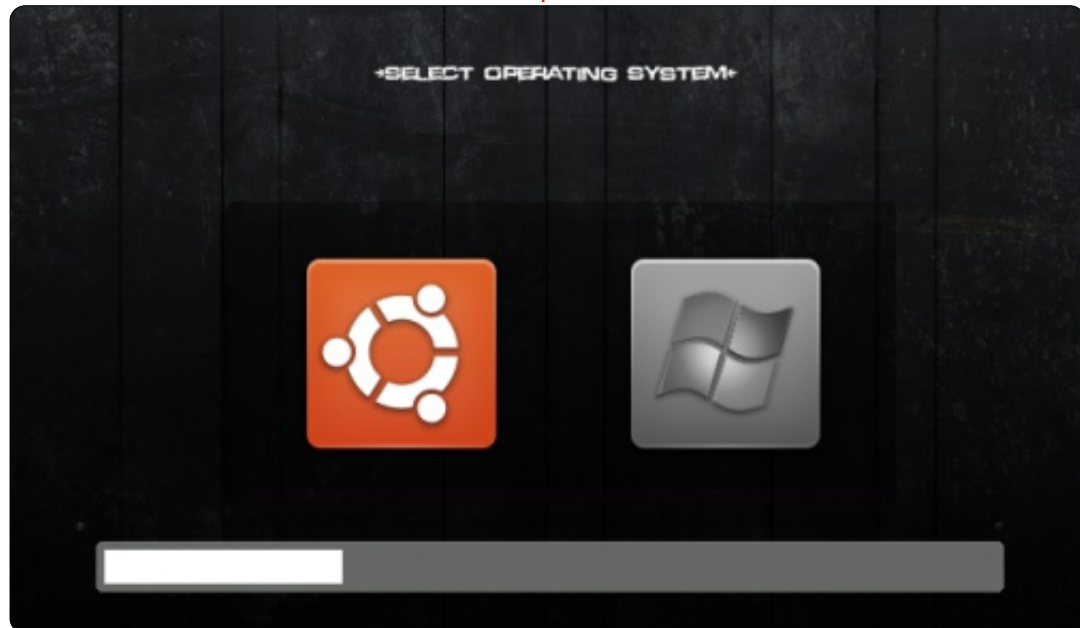
Edit /etc/default/burg and uncomment:

```
GRUB_DISABLE_LINUX_RECOVERY="  
true"
```

Apply the configuration

- update-burg

Here is the screen that will be displayed at startup:



Note: Note that this boot screen doesn't propose recovery mode. The full startup menu is still available on the usb key. You should select to boot on the usb key, and hold shift to access to the following menu.

Note: This screenshot (right) is showing additional boot options. Ref below to see how to install live OS into the usb key.

Create a startup key from a working system

If you have a system running, it is easy to recreate a USB key with the following procedure:

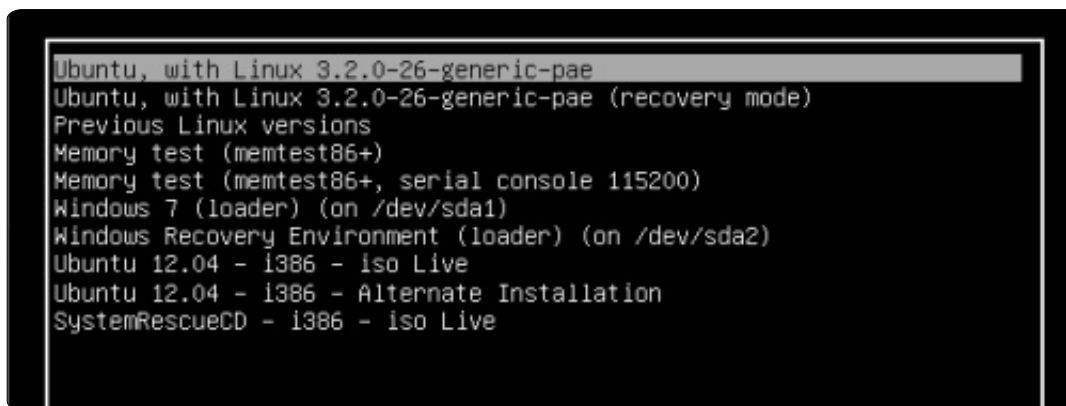
Mount the new key in /media/usb... Note: The key should be prepared with gparted as described above, and have the partition formatted as ext2.

```
mount /dev/sdc2 /media/usb
```

Copy the content of the original key to the new startup key:

```
cp -a /boot/* /media/usb
```

Label the new key to be a boot key:



```
e2label /dev/sdc2 BOOT
```

Install grub2 to the new disk:

```
grub-install --force --no-floppy --boot-directory=/media/usb --root-directory=/ /dev/sdc
```

Note: Each time the kernel is updated, the second key will also have to be updated using this procedure.

Backup the startup key and store it in a safe location

The usb key you just created is now the only way you have to start your computer. It is mandatory to have a backup of it and to be able to recreate it.

Cleanup unused space of /boot

partition:

```
dd if=/dev/zero of=/boot/todelete
```

```
rm /boot/todelete
```

Umount /boot partition:

```
umount /dev/sdb2
```

Backup the mbr of the usb key:

```
dd if=/dev/sdb of=startup.mbr bs=512 count=1
```

Backup the boot partition:

```
dd if=/dev/sdb2 of=startup.sdb2
```

Compress the backup:

```
tar cvjf startup.bkp.tar.bz2 startup.mbr startup.sdb2
```

Store the file

startup.bkp.tar.bz2 into a safe area.

Warning: This to perform an update of the startup key image after every kernel update.

Restore the startup key to another key

The following action should be performed on the computer freshly installed, or from a live version of the OS created from UnetBootIn, or start from the startup key when installation is completed.

Plug the new target usb key and identify its device. Let's assume it is /dev/sdc. Enter these commands:

```
dd if=startup.mbr of=/dev/sdc
```

```
mkfs.vfat /dev/sdc1
```

```
dd if=startup.sdb2 of=/dev/sdc2
```

```
fsck -y /dev/sdc2
```

```
mount /dev/sdc2 /media/usb
```

```
grub-install --force --no-floppy --boot-directory=/media/usb --root-directory=/ /dev/sdc
```


HOWTO - CREATE A THIEF-PROOF COMPUTER

This command could be a little bit long. Executing the following command will let dd write a status of its progress:

```
kill -USR1 $(pidof dd)
```

Store data in a remote location to ensure its availability

Our goal is to store data in a place that will ensure its availability even if the hardware is lost. The easiest solution is to use the cloud services provided by one of the following companies:

Online storages services:

- * 5GB up to 20GB Free - <https://one.ubuntu.com/>
- * 5GB Free - <https://www.wuala.com/>
- * 2GB Free - <https://www.dropbox.com/>
- * 2GB Free - <https://www.spideroak.com/>
- * 5GB Free - <https://drive.google.com/> --> with grive: <https://github.com/Grive/grive>

For Windows only, some additional space can be used to

store non-confidential data:

- * 5GB Free - <https://www.sugarsync.com/>
- * 7GB Free - <https://skydrive.live.com/>

Offline backup service:

- * 25GB Free - <https://www.hubic.me/>

Online notebook:

- * 60MB/month Free - <https://www.evernote.com/>

This solution is not intended to replace real backup/restore solutions, but has the advantages of being cheap and easy to setup.

Ensure the confidentiality of data stored into the cloud

The cloud is a private storage area provided by an external company. This description may not sound correct, because it is mixture of private and external. So, if we consider that this external area is not a fully private place, we will have to add another layer of encryption to secure our data in the cloud. For this, we will use encfs, and we will configure pam to

automatically unlock the directory during the login process.

Install encfs and fuse-utils using the following command as root:

```
apt-get install encfs fuse-utils
```

```
sh -c "echo fuse >> /etc/modules"
```

```
modprobe fuse
```

```
adduser $USER fuse
```

Configure the encryption of sensitive data into in Ubuntu One

Execute the following commands as standard user:

```
sudo apt-get install libpam-mount libpam-encfs
```

```
LC_ALL=C encfs  
/home/$USER/Ubuntu\  
One/.encrypted  
/home/$USER/encrypted/
```

Let encfs create the directories. Select the pre-configured paranoia mode (p), or just press enter to have normal protection.

Enter the passphrase twice (it should be the same as for the account if you want to use pam_mount).

It is possible to automatically decrypt the encfs directory using pam_mount. This will work only if the passphrase of the user account is the same as the passphrase of the encfs directory.

Edit the file /etc/security/pam_mount.conf.xml look for the line <!-- Volume definitions -->. Append the following lines just after by replacing <<user>> with your login

```
<volume user="<<user>>"  
fstype="fuse"  
path="encfs#/home/<<user>>/Ub  
untu One/.encrypted"  
mountpoint="/home/<<user>>/en  
crypt" />
```

Passphrase management

The LUKS encryption system can manage up to eight passphrases (in this article, we already used two). Adding a password can be done with the following command:

HOWTO - CREATE A THIEF-PROOF COMPUTER

```
cryptsetup luksAddKey  
/dev/sda5
```

To delete a passphrase:

```
cryptsetup luksKillSlot  
/dev/sda5 <the slot number to  
be deleted>
```

To change a passphrase, you need to add the new key, then to kill the slot of the key to be replaced.

Encfs has only one passphrase. The passphrase can be changed with the following command:

```
encfstl passwd ~/Ubuntu\  
One/.encrypted/
```

Add live OS into the usb key

On the following section, we will assume that the FAT32 partition of the startup key is mounted in /media/usb. If this is not yet the case, execute the following command to do it:

```
mkdir /media/usb  
mount /dev/sdb1 /media/usb
```

Add ubuntu desktop on the usb

```
#!/bin/sh  
exec tail -n +3 $0  
# This file provides an easy way to add custom menu entries. Simply type the  
# menu entries you want to add after this comment. Be careful not to change  
# the 'exec tail' line above.  
menuentry "Ubuntu 12.04.1 - i386 - iso Live" {  
    set gfxpayload=800x600x16  
    set root=(hd0,msdos1)  
    set isofile="/iso/ubuntu-12.04.1-desktop-i386.iso"  
    search --set -f $isofile  
    loopback loop $isofile  
    linux (loop)/casper/vmlinuz boot=casper iso-scan/filename=$isofile noeject noprompt  
    splash -- locale=fr_FR console-setup/layoutcode=fr  
    initrd (loop)/casper/initrd.lz  
}
```

stick. Create the directory
/media/usb/iso

```
mkdir /media/usb/iso
```

Download *ubuntu-12.04.1-desktop-i386.iso* from <http://releases.ubuntu.com/precise/> and copy it to /media/usb/iso.

Create the file
/etc/grub.d/42_custom with the text shown above.

Upgrade grub with the following command

```
update-grub
```

Add Ubuntu Alternate CD on the usb stick. Download Ubuntu

and copy it in /media/usb/iso.
Create the file
/etc/grub.d/43_custom with the content shown below:

Upgrade grub with the following command

```
update-grub
```

```
#!/bin/sh  
exec tail -n +3 $0  
# This file provides an easy way to add custom menu entries. Simply type the  
# menu entries you want to add after this comment. Be careful not to change  
# the 'exec tail' line above.  
menuentry "Ubuntu 12.04.1 - i386 - Alternate Installation" {  
    set gfxpayload=800x600x16  
    set root=(hd0,msdos1)  
    search --set -f /iso/ubuntu-12.04.1-alternate-i386.iso  
    linux /iso/vmlinuz noeject -- locale=fr_FR  
    console-setup/layoutcode=fr  
    initrd /iso/initrd.gz  
}
```

HOWTO - CREATE A THIEF-PROOF COMPUTER

Add System Rescue CD on the usb stick. Download and copy it in /media/usb/iso. Create the file /etc/grub.d/44_custom with the content shown top right.

Upgrade grub with the following command:

```
update-grub
```

Online security

You are using Linux, this is a first good step for online security. If you intend to use your computer directly connected to the Internet, you should at least start the firewall by executing the following command:

```
ufw enable
```

Remove the key after startup

Warning: In this section, it is proposed to modify the authentication process. An error could block the authentication of your computer. If you do so, start your computer in recovery mode and delete the modification you previously did. Be aware that the option to remove the key is

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries. Simply type the
# menu entries you want to add after this comment. Be careful not to change
# the 'exec tail' line above.
menuentry "SystemRescueCD - i386 - iso Live" {
    set gfxpayload=800x600x16
    insmod ext2
    insmod loopback
    insmod iso9660
    set root=(hd0,msdos1)
    set isofile="/iso/systemrescuecd-x86-2.8.1.iso"
    search --set -f $isofile
    loopback loop $isofile
    linux (loop)/isolinux/rescuecd isoloop=$isofile setkmap=fr
    initrd (loop)/isolinux/initram.igz
}
```

available only in the graphical interface. We do consider that if you start a console, you can as well mount the /boot partition manually.

To not compromise the security of your PC, you need to carry the usb key with you all the time even if the PC is still running and locked.

To make this easy we will configure the computer to automatically mount and umount the key on different occasions:

Umount the key:

- At startup to let you unplug the key and go before login
- When the computer is locked – to

let you unplug the key when you need to leave your PC

- When you close your session

Mount the usb key:

- When you open a session
- When the session is unlocked

Umount key after boot

Add the following as the first active line of the file /etc/rc.local

```
umount /boot
```

This will umount the usb key after boot.

Now, we need to authorize a

standard user to mount and umount the /boot partition which is in the startup key. To do so, it is required to update the /boot description in the file /etc/fstab and append the option 'users' as an option. After modification, the line should look like this:

```
LABEL=BOOT    /boot
ext2 noatime,users
0            2
```

To automatically mount and umount the key when the session is open or close, we will use the capabilities given by PAM. libpam-script will allow us to execute a script when the user opens or closes a session.

HOWTO - CREATE A THIEF-PROOF COMPUTER

Install libpam-script with the following command:

```
apt-get install libpam-script
```

/usr/share/libpam-script/pam_script_ses_open is executed when the session is open, and will mount the partition /boot. Create this script with the following content:

```
#!/bin/bash
if [[ "$PAM_USER" = "lightdm"
]] || ( mount | grep /boot >
/dev/null 2>&1 ); then
    exit 0
fi
if ( mount /boot > /dev/null
2>&1 ); then
    /usr/bin/aplay
/usr/share/sounds/purple/receive.wav > /dev/null 2>&1
fi
exit 0
```

This script mounts the /boot partition, and plays a sound letting you know that the key has been successfully reconnected.

Change permissions to make it executable:

```
chmod 755 /usr/share/libpam-script/pam_script_ses_open
```

/usr/share/libpam-

script/pam_script_ses_close is executed when the session is closed, and will dismount the partition /boot. Create this script with the following content:

```
#!/bin/bash
if [[ "$PAM_USER" = "lightdm"
]]; then
    exit 0
fi
device=$(mount | grep /boot |
cut -c -8)
if ( umount /boot > /dev/null
2>&1 ); then
    umount $device > /dev/null
2>&1
    /usr/bin/aplay
/usr/share/sounds/purple/send.wav > /dev/null 2>&1
fi
exit 0
```

This script umounts the /boot partition and all the partitions of the usb key, and then plays a sound letting you know that you can remove the usb key safely.

Change permissions to make it executable:

```
chmod 755 /usr/share/libpam-script/pam_script_ses_close
```

We need now to add pam_script to the session management. Modify the file /etc/pam.d/lightdm and append the pam_script line

below just after the line @include common-account:

```
@include common-account
```

```
session optional
pam_script.so
```

We will now create a script managing the screen-saver. The script below is applicable to xscreensaver which is the default screensaver of xubuntu. If your screensaver is different, you can replace it by xserver, or (more difficult), you can update the script below.

Create the file /usr/local/bin/startup_key_manager.sh with the following content:

```
#!/bin/bash
/usr/bin/xscreensaver-command
-watch | while read line; do
    if [ x"${echo "$line" |
grep 'LOCK'}" != x ] ; then
        /usr/share/libpam-script/pam_script_ses_close
    fi
    if [ x"${echo "$line" |
grep 'UNBLANK'}" != x ] ;
then
        /usr/share/libpam-script/pam_script_ses_open
    fi
done
```

Change the permissions to make it executable:

```
chmod 755
/usr/local/bin/startup_key_ma
nager.sh
```

This script will monitor the screensaver and manage the /boot partition of the key accordingly.

This script should be added to start automatically when the session is open. As a normal user, create the script ~/.config/autostart/startupKeyManager.desktop with the following content:

```
[Desktop Entry]
Encoding=UTF-8
Version=0.9.4
Type=Application
Name=startupKeyManager
Comment=startup key manager
Exec=/usr/local/bin/startup_k
ey_manager.sh
StartupNotify=false
Terminal=false
Hidden=false
```

The script will be activated when you will start another session.

Warning: When you are updating the kernel, be sure the session will stay open and the screen server stays deactivated during the upgrade.

Two-factor authentication

Warning: In this section, it is proposed to modify the authentication process for graphical as well as console login. An error could block the authentication of your computer. If you do so, start your computer in recovery mode, and delete the modification you previously did.

We have now a system which is secured and easy to use, but we can improve the security a little bit by adding a two-factor authentication requiring the USB key to be plugged and the password to be correct before opening the session. With this two-factor authentication, you will be sure that, in the event that somebody knows your password, she will not be able to unlock your session when you are at the coffee corner with the USB key in your pocket.

To activate the two factor authentication, we will use the pam module previously installed: pam-script.

The logic would be to use

pam_usb, but this approach is inconvenient: It requires an action on every USB key you have, and makes key replication more complex. So, we will authenticate the usb key based on the keyfile present in it using pam_script.

Create the script /usr/share/libpam-script/pam_script_ses_auth dedicated to authenticate the usb key with the following content:

```
#!/bin/bash
mount /boot
result=1;
if ( shasum -c --status
/usr/share/libpam-
script/keycheck ); then
    result=0
fi
umount /boot
exit $result
```

Make it executable:

```
chmod 755 /usr/share/libpam-
script/pam_script_auth
```

The sha1sum is used to validate the key; the keycheck file is created with the following commands:

```
shasum /boot/keyfile >
/usr/share/libpam-
script/keycheck
```

```
chmod 444 /usr/share/libpam-
script/keycheck
```

We need now to add pam_script into the system authentication process just after the authentication by password. Modify the file /etc/pam.d/common-auth and add pam_script just after pam_deny as follow:

```
auth requisite
pam_deny.so
```

```
auth required
pam_script.so
```

Troubleshooting

In some circumstances you may need to access to the data of the encrypted partition without booting the computer. Here are some methods to do so.

Boot in recovery mode

Boot from the usb stick and select recovery mode. Select root, Drop to root shell prompt. Mount / with a read write access and mount /boot using the following commands

```
mount -o remount,rw /
```

```
mount /dev/sdb2 /boot
```

Note: As you are using your startup key to boot your PC, you are identified to be the owner of the machine if you used a keyfile; you will be granted root access without any password. Without the key, such a startup is impossible and you will have to follow the instruction of the next section to access your data.

Manual access to the partition

To access to the encrypted partition, boot on a live Operating System and follow the procedure below to mount and umount the disk.

Mount encrypted partition

```
modprobe dm-crypt
```

```
cryptsetup luksOpen /dev/sdb5
crypt
```

(Enter your passphrase)

```
vgscan --mknodes
```

```
vgchange -ay
```

```
mkdir /mnt/crypt
```

HOWTO - CREATE A THIEF-PROOF COMPUTER

```
mount /dev/VolGroup/LV_slash
/mnt/crypt
```

Umount partition

```
umount /mnt/crypt
```

```
vgchange -an
```

```
cryptsetup luksClose crypt
```

To gain access to the encrypted partition from (initramfs):

Mount encrypted partition

```
cryptsetup luksOpen /dev/sdb5
crypt
```

```
mkdir /mnt/crypt
```

```
mount /dev/VolGroup/LV_slash
/mnt/crypt
```

Umount partition

```
umount /mnt/crypt
```

```
cryptsetup luksClose crypt
```

Reinstall the secure system and keep data in the home directory

In case of a major issue, you may have to reinstall your system from scratch. Boot on Ubuntu

Alternate image

- Enter the name of the computer
- Enter the full name of the main user
- Enter the username for the account
- Choose a password and enter it twice
- Do not select to encrypt home directory (We will reuse the encrypted partition already existing into the system)
- Set clock timezone
- Partition disks: Manual
- Select: Configure encrypted volume
- Keep current partition layout

and configure encrypted volumes:

Yes

- Activate existing encrypted volume
- Enter the passphrase
- You will see the LVM volumes in the disk partition description
- Define mounting point as described in the previous chapter (Format the partition / and /boot but do not format the partition /home)
- Install the OS
- Reboot

After this installation /boot and / have been recreated from

```
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

LVM VG VolGroup, LV LV_home - 162.6 GB Linux device-mapper (linear)
#1          162.6 GB      f  ext4      /home
LVM VG VolGroup, LV LV_slash - 50.0 GB Linux device-mapper (linear)
#1          50.0 GB      f  ext4      /
LVM VG VolGroup, LV LV_swap - 2.0 GB Linux device-mapper (linear)
#1          2.0 GB      f  swap      swap
Encrypted volume (sda5_crypt) - 214.6 GB Linux device-mapper (crypt)
#1          214.6 GB      K  lvm
SCSI1 (0,0,0) (sda) - 268.4 GB ATA VBOX HARDDISK
#1 primary  53.7 GB      B      ntfs
#2 primary  127.9 MB      ext2
#5 logical  214.6 GB      K  crypto  (sda5_crypt)
#4 primary  16.8 MB      fat16
SCSI2 (0,0,0) (sdb) - 8.6 GB ATA VBOX HARDDISK
#1 primary  126.9 MB      B  K  ext2      /boot
#5 logical   8.5 GB
```

Undo changes to partitions

Finish partitioning and write changes to disk

scratch. It is then required to reapply the configuration described in the previous section. If you use a keyfile to unlock the secured partition, this file should be reinstalled in the /boot partition from the backup you did. If you previously saved the installed packages into a file installed-packages as described in the Security: section, it is possible to reinstall them with the following commands:

```
apt-get update
```

```
apt-get upgrade
```

```
dpkg --set-selections <
installed-packages
```

```
apt-get -u dselect-upgrade
```

To go further and improve the security and data integrity

Additional action could improve the security of your computer. You can, for instance, remove Windows from your computer. In this case, your computer will not boot at all without the startup key, and will be unusable and without any evidence that it holds an encrypted partition. Then you can add a BIOS

HOWTO - CREATE A THIEF-PROOF COMPUTER

password to avoid booting from usb, and add a password to burg and grub to avoid startup command modification.

You can also use TrueCrypt with/without inner volume to secure confidential data.

To go even further, you can apply the recommendation from the NSA:
<http://www.nsa.gov/ia/ files/os/re dhat/rhel5-guide-i731.pdf>

References

This article has been written based on information found in the Internet:

<https://help.ubuntu.com/community/EncryptedFilesystemHowto>

<http://ubuntuforums.org/showthread.php?t=1549847>

<http://ubuntuforums.org/showthread.php?t=1369019>

<http://blog.stalkr.net/2012/05/usb-rescue-and-secure-boot-disk.html>

<http://askubuntu.com/questions/63594/mount-encrypted-volumes-from-command-line>

<http://ubunteros.tuxfamily.org/spip.php?article204>

<http://doc.ubuntu-fr.org/tutoriel/chiffre son disque>

<http://doc.ubuntu-fr.org/cryptsetup>

<http://doc.ubuntu-fr.org/encfs>

<http://linuxconfig.org/linux-authentication-login-with-usb-device>

<http://artisan.karma-lab.net/petite-introduction-a-pam>

<http://www.psychocats.net/ubuntu/security>

<https://code.google.com/p/cryptsetup/>



Xavier Berger is a specialist in Linux and network deployment who enjoys hiking, geocaching, skiing, and spending time with his family. His web site is: <http://xberger.free.fr>

16x16 SUDOKU

Numbers 0 to 9 and letters A to F are to be filled into the 16x16 grid so that every row, every column, and every 4x4 box contains 0 to 9 and A - F.

1	4	5			B	3		E							7
	F		0		1		9	2			4	5		D	E
		C		F	0				8		5	9	6		4
	B	D	9		E				C		F	8		1	
	1	4	C		3	B	A	8					D		
						4					D	3	B	6	9
		2	F				0			3	B				1
B	3			2					1		C			A	
	9			A		8					0			4	C
F				3	4			6				0	1		
2	8	0	4	9					A						
		1					E	4	3	5		A	9	F	
	2		3	C		F				4		6	E	9	
C		6	7	4		E				A	3		F		
E	0		1	B			3	D		F		4		C	
4							6		5	C			8	3	2

Solutions are on the second last page.

Puzzles are copyright, and kindly provided by,
The Puzzle Club - www.thepuzzleclub.com



HOW-TO

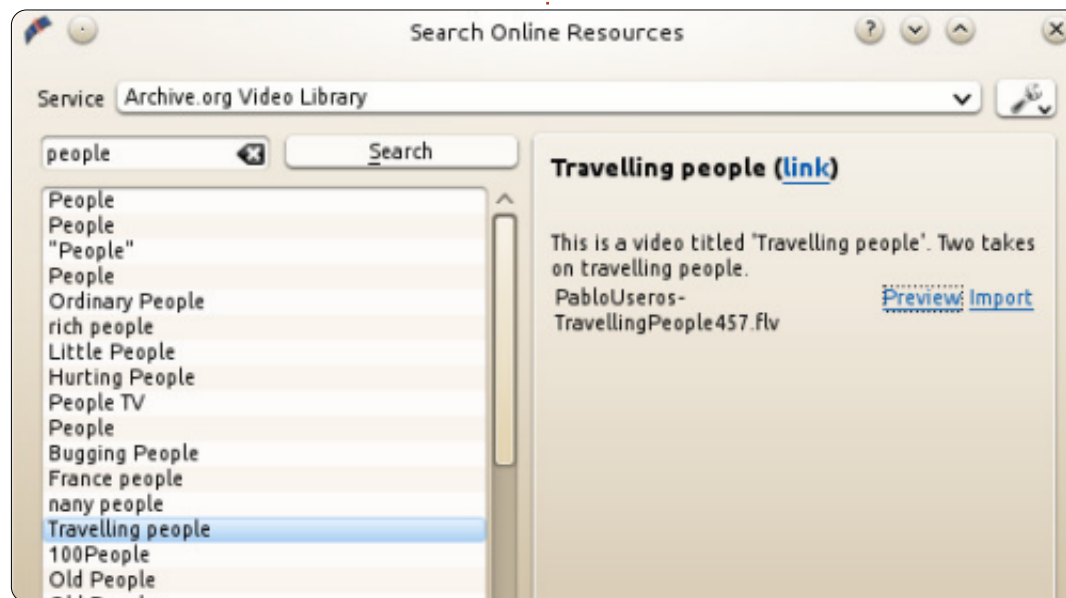
Written by Ronnie Tucker

Video Editing With Kdenlive - Part 4

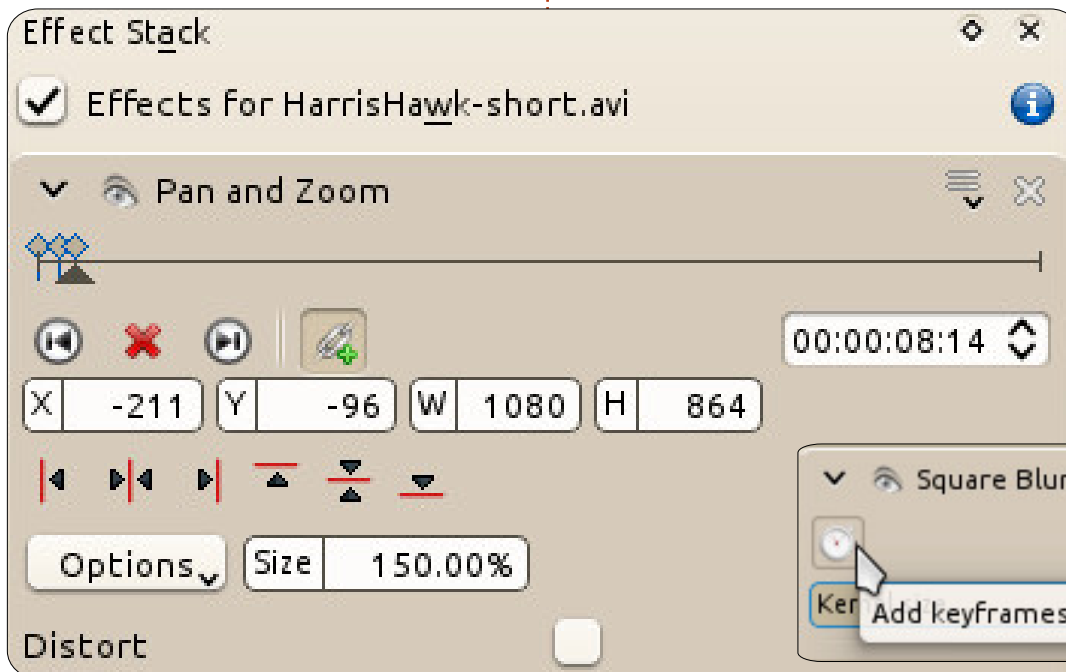
Although we've touched on effects that you can apply to a clip, we'll go a bit more in-depth on clips in this article.

First, let's import a clip. Another way to add a clip is to right-click in the clips area and choose 'Online Resources' – which will give you a search window of free clips to choose from. The dropdown menu lets you choose from audio/video/graphic libraries, and you can enter some text to filter the list of things to import.

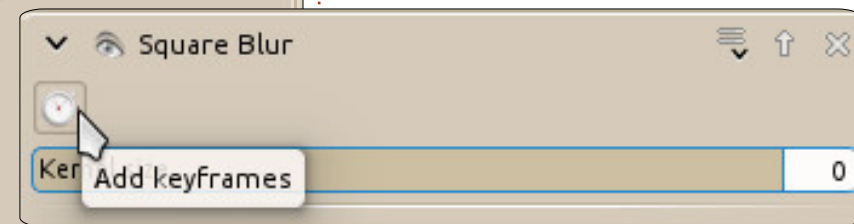
I'm going to import my own video clip, and apply some effects to it. The first effect I'm going to apply is Add Effect > Crop and Transform > Pan and Zoom. This is an effect which can be animated, which is something we touched on briefly last issue. Just like last time, we add a keyframe and apply some settings. Do that several times and you have an animated effect. So, what I've done is made a keyframe 5 seconds in, and another at 8 seconds in.

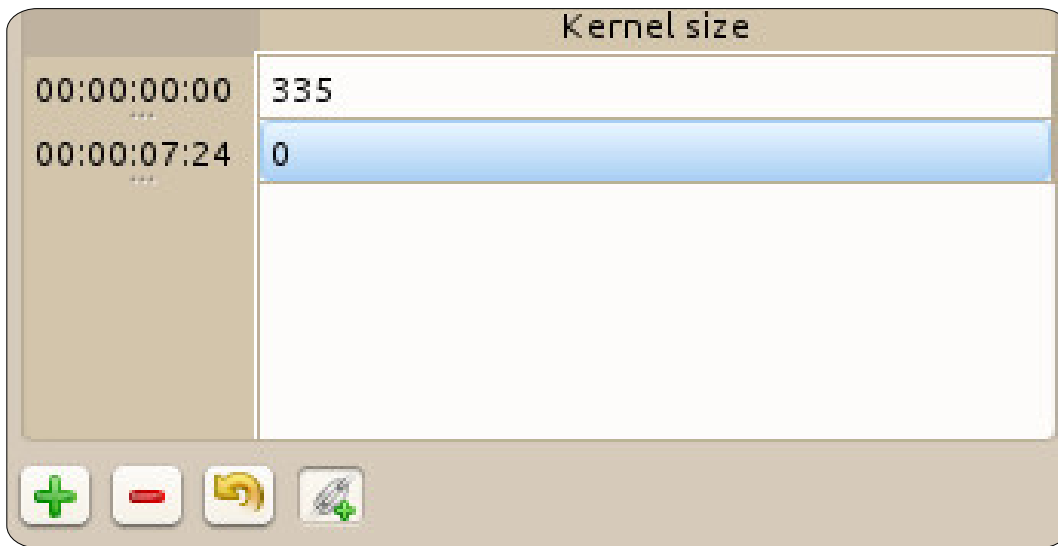


First, click the 'synchronise with timeline' button (chain links) to move your effect slider with the timeline slider. I've not touched the settings for the 5sec keyframe, but I've upped the zoom level on the 8sec keyframe, and adjusted the X and Y values to center the subject a bit. What this means is that the video plays normally for 5secs, then, from the 5sec mark to the 8sec mark, it'll spend those three seconds zooming in to my 8sec keyframe. Of course, you don't have to zoom, you could just as easily track your subject without the zoom.



The only problem with effects is that not all of them can be animated. The best gauge I can give is to look through the effects available, and, if it says 'keyframeable,' then it can definitely be animated. Some effects can have keyframes, but





you'll have to apply them to see if they'll do keyframes. If they can, they'll have a small icon of a stopwatch in the effect properties:

This blur effect can be animated by clicking the stopwatch to apply a keyframe.

TIP: Clicking the eye icon beside the effect name will hide it, while clicking the down-arrow will hide the properties of the effect until you click the arrow to reshow them. Both are useful when you have several effects taking up space or CPU time.

If the time on a keyframe is wrong, you can click on the keyframe and set the time on it

using the 'Position' slider. In this case, 'Kernel size' is the amount of blur. In this blur example, I have it really blurry at the start (00:00:00:00), and getting sharper, until it's returned to normal at roughly the 7sec mark. I still have my zoom and pan too.

Next issue we'll start a new series of Blender tutorials!



Ronnie is the founder, and editor, of Full Circle, an official Ubuntu member, and part-time artist whose work can be seen at: <http://ronnietucker.co.uk>

CODEWORD

Every number in the grid is 'code' for a letter of the alphabet. Thus the number '2' may correspond to the letter 'L', for instance. All - except the difficult codeword puzzles - come with a few letters to start you off

15	21	12	26	8	10	12	20		8	14	11	21													
	23		15		12		2		26		2														
1	5	26	1	8	1		15	21	11	9	5	21													
	7		2		21	24	21		15		26														
1	2	6	12		15		1	12	2	2	3	21													
	4				11				12																
22	21	25	10	11	1		16	14	1	16	21	4													
			12				21				13														
1	16	9	21	1	11		5		5	15	21	9													
	26		23		15	14	11		10		17														
17	10	17	26	4	26		26	4	2	5	11	1													
	18		17		8		12		14		2														
19	14	10	11		5	15	21	1	1	14	15	21													
1	2	3	4	5	6	7	8	9	10	11	12	13													
			D							T															
14	15	16	17	18	19	20	21	22	23	24	25	26													
			C																						
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Solutions are on the second last page.

Puzzles are copyright, and kindly provided by, **The Puzzle Club** - www.thepuzzleclub.com

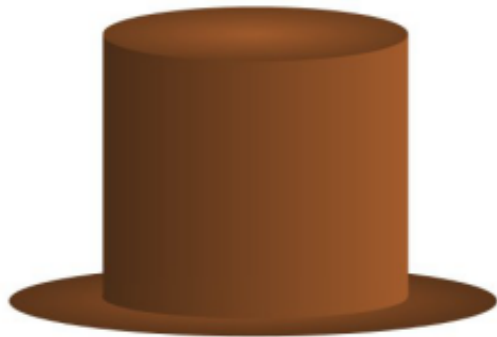


HOW-TO

Written by Mark Crutch

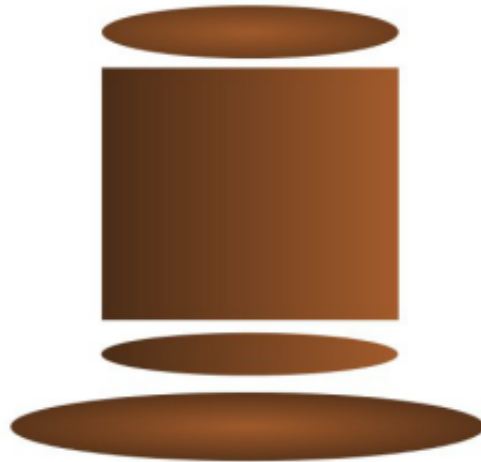
Inkscape - Part 7

At the end of last month's article, I promised that we would take a look at the too pristine hat that's adorning our now-lumpy snowman. The hat was added way back in the second instalment of this series, when all we could draw were ellipses and rectangles, and is a good example of how a few carefully chosen objects can fool the eye into seeing shapes that aren't really there. Here's how it looks when removed from the snowman's head and rotated a little:



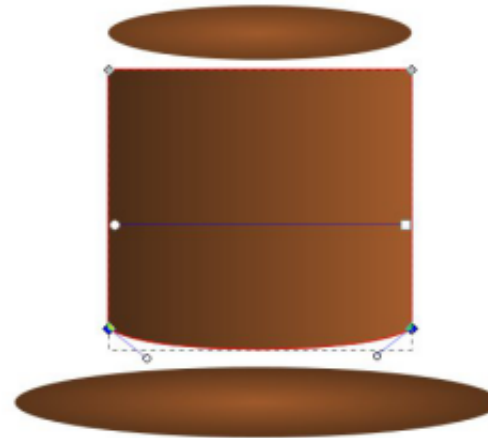
There's obviously an ellipse at the top, and another for the brim, but what of the main body of the hat? With your new knowledge

from last month, it would be easy to create by just converting a rectangle to a path, and curving the top and bottom segments, but this version pre-dates that insight by several months. Exploding the hat into its constituent parts makes things a little clearer:



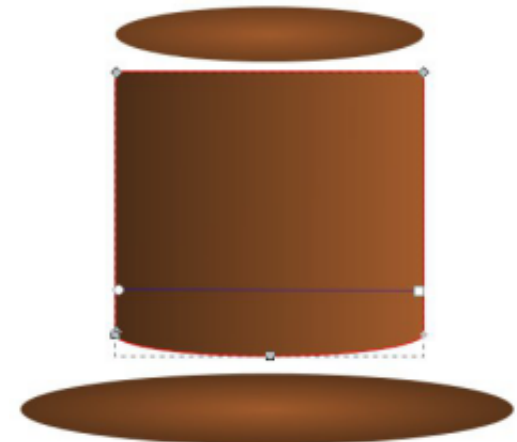
The body is just a rectangle with an ellipse peeking out from beneath it. By giving them matching gradients we could create the illusion of a single object. The curve at the top doesn't exist at all, it's just an optical effect created by putting the rectangle behind the ellipse that forms the top of the hat.

Fooling the eye for artistic effect can be very useful, but, in this case, it was really just a workaround for our lack of Inkscape skills. It would be better if the body of the hat was just one object with one gradient. Our complete hat would then consist of an ellipse for the top, an ellipse for the brim, and a path for the body. We could do that by converting the rectangle to a path and changing the bottom edge to a Bézier curve:



But we've already got the shape we want in the outline formed by the rectangle and ellipse. What we really want to do is to combine

those objects into a single path. In order for Inkscape to know which items we want to combine, our first step is to select them both. Switch to the selection tool, click on the rectangle, and then hold down SHIFT while clicking on the ellipse. The status bar should tell you that you have "2 objects of types Rectangle, Ellipse" selected. If not, use Edit > Deselect (or just click on the background, away from any other objects) in order to clear your selection, then try again. Once you're happy that you've got the right two objects selected, use the Path > Union menu entry (CTRL+) to combine your objects into a single path:



If you zoom in closely you'll notice that our new path isn't quite as perfect as we would like. Some misalignment of the ellipse to the rectangle has resulted in extra nodes being created that we'll need to manually edit using the Node tool. Creating paths from other objects in this way can be a fast way to get the basic outline of a shape, but usually you'll need to do some manual editing afterwards.



As well as adding shapes together to form a single path, we can also subtract them, cutting one shape out from another. Looking at the top of the hat, we could use the ellipse to cut away a curved section. You don't need to be a mathematical genius to understand that 1-2 doesn't give you the same result as 2-1.

Similarly subtracting a rectangle from an ellipse won't give you the same result as subtracting an ellipse from a rectangle, so Inkscape needs some way of knowing which object is the one we're removing, and which is the one we're removing it from. This is done using the stacking order.

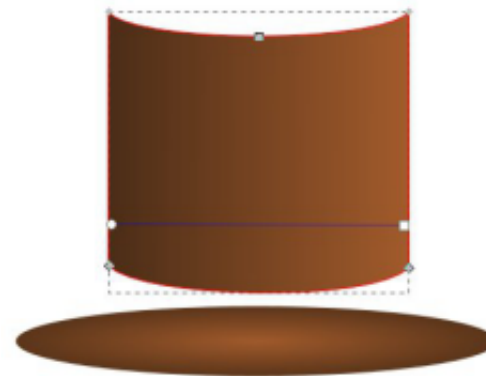
The stacking order was briefly mentioned back in the first part of this series: objects within Inkscape can sit 'on top of' each other, with ones at the top of the stack obscuring those below (assuming they're all fully opaque). When you've got an object selected using the Selection tool you can move it up and down the stack using these four buttons on the tool control bar:

To cut one object from another, the shape that is being removed



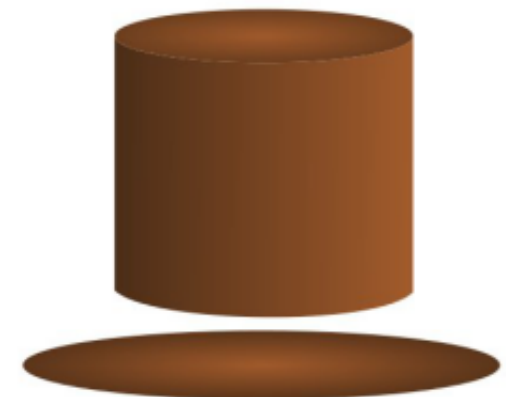
must be on top of the other object – in our case the ellipse needs to be on top of the rectangle-with-a-curved-base. It's probably already on top in the stack, but it doesn't hurt to get into the habit of always moving your cutting object on top

of the object to be cut. This is easy to achieve by selecting the ellipse and using the last of the four buttons to bring it right to the top of the stacking order. Now you need to position the ellipse to cut out the shape you want – in this case overhanging the edge of the rectangle. With that done it's a simple case of selecting both objects and using Path > Difference menu entry (or CTRL and the minus key) to subtract the ellipse from the rectangle:



Unfortunately there are a couple of problems with the result: the first is that, once again, we have some rogue nodes that will need to be manually tidied up; the second is that our ellipse has now completely disappeared! That's a side effect of the Path > Difference operation: the object you are cutting out is also removed in the process.

Press CTRL-Z (or use Edit > Undo) to restore your ellipse back to its pre-cut status. Now select just the ellipse and use Edit > Duplicate (CTRL-D). This will create a copy of the selected object(s) at exactly the same location on the canvas, but at the top of the stack. It's worth getting used to the keyboard shortcut for this as it's a particularly useful feature – it not only gives you a disposable copy of your cutting object to use, but also saves you having to manually move it to the top of the stack. It even leaves the new object selected, so you just have to SHIFT-click on the object to be cut, then use Path > Difference. Now our former rectangle has a curved base and a curved top, and we still have an ellipse to form the lid.



But still there's a problem. A close look at the join between the curved rectangle and the lid shows a thin line showing through from the canvas or objects below. Inkscape uses "anti-aliasing" when it draws your objects. This tries to approximate edges that don't fall precisely on pixel boundaries by drawing a thin, translucent line to give the visual impression of a smooth curve. Usually it works well, but when two such boundaries meet it can lead to a slight gap where you don't really want one.

There are a couple of solutions to this anti-aliasing issue: we could add a stroke to one or both of the objects to cover up the gap, but that's not always appropriate and doesn't help if the problem objects already have strokes; alternatively we could avoid relying on our objects perfectly abutting one another by making one of them overlap the edge of the other. In this case we don't really need the rectangle to have a curved top – we already had the visual effect we were looking for with the ellipse sitting on top of the rectangle.

Let's CTRL-Z back to our

straight-topped hat. It may seem that we've gone round in circles, but, along the way, you've learned about Path > Difference, the visual issues that anti-aliasing can sometimes cause, and the fact that sometimes it's better to draw something that looks right, even if its underlying structure isn't as pure or correct as you might like. Before finally reassembling the hat it might be worth adding some creases and crumples using the path tools from last month:



While you were in the Path menu looking for Union and Difference, you probably noticed the other entries grouped alongside them: Intersection, Division, Exclusion and Cut Path. Together all these form the "Boolean" path operations – named after 19th Century mathematician George Boole who formulated the rules for combining binary data that underlie these operations. They all require two objects (though Union can work with more than two), with the result usually differing based on

Original Objects		
Union		
Difference		
Intersection		
Exclusion		
Division		
Cut Path		

which of the objects is higher in the stack. The above table demonstrates the effect of applying each of these operations to an overlapping square and circle, and shows the difference in results depending on which of these objects is on top of the other (in the case of Division and Cut Path the resultant objects have been separated slightly to make it easier to see the effect of these operations).

With a bit of practice these Boolean operations will allow you to produce complex paths from a few simple shapes. They provide a

fast way to produce rough initial outlines that you can then tweak with the node tool into something more refined. But don't dismiss them solely as a blunt instrument for coarse construction work: they can also be used to sculpt and shape with the precision of a scalpel.



Mark has been using Linux since 1994, and uses Inkscape to create two webcomics, 'The Greys' and 'Monsters, Inked' which can both be found at: <http://www.peppertop.com/>



As our CRUD (Create, Read, Update, Delete) continues, we are going to jump right on in. I want to show you how to use JavaScript itself before we rewrite the app with JQuery. Understanding the language is more important than how to use the libraries.

One of the first functions we will write is going to be a very well used one. You might actually want to save this for future use. The function needs to look at the document object, search for ids, and return the one you are looking for. First, we start out by creating a function that takes one argument, we'll call it ge for "get element". To create a function you simply write:

```
function ge(id) {
    // code here
};
```

Now that we have a function, we need to write the code. We are searching the document object model for an id, which the

document object has a function for. So let's bring the results of that function into a variable and return it.

```
// gets an element by its id
// from the document object
// model (DOM)

function ge(id) {

    var theElement =
    document.getElementById(id);

    return theElement;
};
```

The line that starts with // is called a comment. This line is to help coders keep track of what is going on in their code, and allow them to leave notes for other people using their code.

The next thing is that we need to handle the form when the submit button is pressed. First, we need to give an id to the form itself. Let's call it "ubuVersForm". So now, the open form tag should look like this: <form id='ubuVersForm'>. We need to add just one more id to our HTML

form. Our submit button should have an id of submit (' id='submit' '). Your form button should now look like this: <input type="submit" name="submit" id='submit' value="Add" />.

So now that we have our form set, we can move back to the JavaScript. The JS contains in its first line an alert to tell us that our JS file is connected. This is pretty annoying when we are testing, so let's change that to a console log—so we can see it if we want to, and forget about it if we don't. Next, we create a variable that contains the form element we are going to be using and manipulating later; let's call that 'form'. Hint: we created a function to get elements by id a little earlier in this article.

So, coding is a lot like logical thinking. Think of how you would tell yourself what needs to happen. If the submit button is pressed, then do process the form. That means we are going to need to have something that attaches a function (let's call it processForm) to the button. Browsers can be

tricky, and there are a few different ways to code some things, and I want to show you good uses of if/else coding... so, we are going to link the function to the button two different ways, but only once. Sounds funny, but it will all make sense.

Let's start out by creating an if/else statement:

```
// Handle form event

if (argument) {

    // do something

} else {

    // do something else

};
```

If statements look very easy, but it's also easy to lose track of what is going on, so commenting these heavily from the start is going to save you headaches down the road. I know I want to fall back on the old and reliable addEventListener, this (next page, top left) will then be in our else statement.

```
// Handle form event
if (argument) {
    // do something
} else { // if not then lets just add the ol' event listener
    // when submit is triggered, run the processForm function
    form.addEventListener("submit", processForm);
};
```

Earlier, I had you create a variable that holds the form element. It should have looked something like this: `var form = ge('uBuversForm');`. This is very useful now, because we can see if the browser let us look at what we can do, what it contains, etc. We are going to look for an `attachEvent` object within our form element (below). If the browser allows it, let's use it.

We are almost ready to try everything out, only one problem now. We need to write the `processForm` function. No sense in adding it to the submit button if it

doesn't exist, right? Start by creating a new function called `processForm` (above right), with one argument, and make it return `false`. Now that you have a function, the first thing we want to happen is to stop all default actions the browser knows to take. Again, we only want to do this if the browser allows for it. To do this we use a object called `preventDefault`.

The above code pulls in the form element (as `formElement`) that you had tied to the event, checks if the browser allows for `preventDefault`, and does

```
// process forms
function processForm(formElement) {
    if (formElement.preventDefault) {
        formElement.preventDefault();
        console.log('form has been processed.');
```



```
    // Return false to prevent the default form behavior
    return false;
};
```

`preventDefault` if it does. The `if` statement is on one line in this case because it is such a simple check and run. You could put this on multiple lines, and add an `else` statement, and maybe console log, if it doesn't allow `preventElement`. You could also add console logs to the `if/else` statement to attach the function to the submit event.

We covered quite a bit of stuff this month. The following is an update of how your code should look now. Thank you for following along, I would love to see what you guys have now, or

questions/comments that you might have. Feel free to share via twitter [@aliendev2](#) and/or hashtag [#FCMWebDev](#).

The full code is on PasteBin at: <http://pastebin.com/pV6RGmBq>

```
// Handle form event
if (argument) { // if the browser allows for attachEvent
    // attach processForm function to trigger when submit button is pressed
    form.attachEvent("submit", processForm);
} else { // if not then lets just add the ol' event listener
    // when submit is triggered, run the processForm function
    form.addEventListener("submit", processForm);
};
```



Michael Youngblood has been in the industry of web design and development for 13 years. He has been working for a world wide wireless tech corp for six years and is working on his bachelor's of science in mobile development.



HOW-TO

Written by Ronnie Tucker

UPDATED!

Write For Full Circle Magazine

Guidelines

The single rule for an article is that **it must somehow be linked to Ubuntu or one of the many derivatives of Ubuntu (Kubuntu, Xubuntu, Lubuntu, etc).**

Rules

- There is no word limit for articles, but be advised that long articles may be split across several issues.

- For advice, please refer to the **Official Full Circle Style Guide:** <http://url.fullcirclemagazine.org/75d471>

- Write your article in whichever software you choose, I would recommend LibreOffice, but most importantly - **PLEASE SPELL AND GRAMMAR CHECK IT!**

- In your article, please indicate where you would like a particular image to be placed by indicating the image name in a new paragraph or by embedding the

image in the ODT (Open Office) document.

- Images should be JPG, no wider than 800 pixels, and use low compression.

- Do not use tables or any type of **bold** or *italic* formatting.

If you are writing a review, please follow these guidelines :

When you are ready to submit your article please email it to:

articles@fullcirclemagazine.org

Translations

If you would like to translate Full Circle into your native language please send an email to ronnie@fullcirclemagazine.org and we will either put you in touch with an existing team, or give you access to the raw text to translate from. With a completed PDF, you will be able to upload your file to the main Full Circle site.

REVIEWS

Games/Applications

When reviewing games/applications please state clearly:

- title of the game
- who makes the game
- is it free, or a paid download?
- where to get it from (give download/homepage URL)
- is it Linux native, or did you use Wine?
- your marks out of five
- a summary with positive and negative points

Hardware

When reviewing hardware please state clearly:

- make and model of the hardware
- what category would you put this hardware into?
- any glitches that you may have had while using the hardware?
- easy to get the hardware working in Linux?
- did you have to use Windows drivers?
- marks out of five
- a summary with positive and negative points

You don't need to be an expert to write an article - write about the games, applications and hardware that you use every day.



Online
BACKUP

Secure
SYNC

Easy
SHARING

Whether you need to access a document you have stored on a remote server, synchronize data between a Mac, Windows or Linux device, share important business documents with your clients, or just rest easy knowing all of your data is safely, securely, and automatically backed up - SpiderOak's free online backup, online sync and online sharing solution can handle all your needs!

SpiderOak offers a different approach to online backup by combining a suite of services into one consolidated tool - free online backup, synchronization, sharing, remote access, and storage. This difference is further measured in our zero-knowledge privacy policy - the first one ever employed in this setting. Our flexible design allows you to handle data from any operating system (Mac, Windows and Linux) or location (external drives, network volumes, USB keys, etc...) using just one centralized account.

- ➔ Access all your data in one de-duplicated location
- ➔ Configurable multi-platform synchronization
- ➔ Preserve all historical versions & deleted files
- ➔ Share folders instantly in web ShareRooms w / RSS
- ➔ Retrieve files from any internet-connected device
- ➔ Comprehensive 'zero-knowledge' data encryption
- ➔ 2 GBs Free / \$10 per 100 GBs / Unlimited devices

<https://spideroak.com>

Download mobile clients
for **iOS & Android**

JOIN SPIDEROAK NOW
Get 2 Free GBs

Get 25% off any SpiderOak package
with the code: **FullcirclemagFans**



ASK THE NEW GUY

Written by Copil Yáñez

Hi, everyone! Welcome back to Ask the New Guy!

If you have a simple question, and want an answer that doesn't require grad school, contact me at copil.yanez@gmail.com.

Today's question comes to us from anonymous poster Schmopil Schmanyey who asks:

Q: How do I make my desktop look like a Terminator's heads-up display?

A: Good question, Schmopil! And may I add that I can tell by your question you are one handsome devil.

If you're reading my column, then you're either an Ubuntu neophyte like me, or you're just a masochist who loves seeing the English language butchered in novel ways. If you're the former, and using Ubuntu is a relatively new experience, then you might not understand the appeal of

tweaking your desktop to make it look like it's been taken over by the ghost of TRON.

But you will, trust me. See, using Ubuntu is like playing a video game. The main campaign is usually fun and exciting, but there are also achievements to be unlocked along the way.

The first achievement to unlock is, of course, installing Ubuntu. Hopefully this magazine, and the plentiful resources online, have shown you how drop-dead simple installation can be. Let's put it in Manspective (the uniquely male point of view that is three parts action movie, one part porn, zero parts reality): installing Ubuntu is akin to showering before a date. Yes, it takes a little bit of time. Yes, most people proceed in a similar fashion (hair, pits, ugly bits). And yes, the process can go horribly wrong if you're not careful. Instead of coming out smelling like an Irish hillside, and looking like a well-kept arboretum, you can come out covered in your own blood and begging for death. I hear you

saying, "That will. Never. Happen. To me." Well, I thought the same thing until this morning. Just sayin'.

Once that main achievement is unlocked, the vast majority of first-time users proceed like a dragon-hunter in Skyrim. Learn The Way of the Voice? Check. Gain Elder Knowledge? Check. Defeat Alduin? Check and done. Ubuntu is the same way. Install Ubuntu? Check. Compose Mail? Check. Open Browser? Check and done.

Unless you have a problem with a scanner, or can't print, your Ubuntu campaign will be worry-free, and you'll have no reason to check in with the forums or ask questions of barely literate columnists in otherwise outstanding magazines.

But just off the main campaign, there are less-traveled avenues to explore, backcountry roads where the local dialect is spoken in banjo, and moonshine is the coin of the realm. In the Skyrim example, the side-quests let you get married,

build a house or adopt a child.

(A quick note: Skyrim's side-quests are what happens when you leave forty-year-old programmers unchecked. I half expect there to be a quest to maximize my retirement holdings.)

Ubuntu has its own hidden achievements; side-quests that take advantage of your growing skill. Schmopil is asking about one of the most popular unlockable achievements in Ubuntu: the customized desktop.

You can find examples all over the web. You can also see some amazing desktops just by turning to the "My Desktop" pages of this magazine. Look at those beauties! There's so much eye candy I want to stick myself with an insulin pen!

But what's really amazing is how easy it is to do some basic tweaking that will make your friends and family think you've secretly joined an evil society bent on world domination, and your humble desktop is the command



center.

Here's how.

First, let's show you a "Before" image of my desktop (right).

Look at it. All sad and boring like me in high school. But not to worry, we're about to give this bad boy a Silkwood shower!

First, let's add a more interesting background, one worthy of the souped up, practically sentient operating system we're working with.

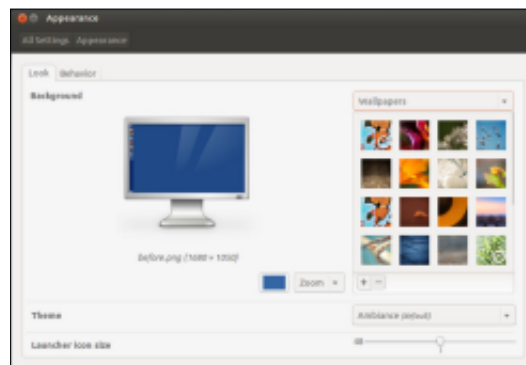
Go online and find an image you like. It can be anything. Maybe you're really into cars. Or rocket ships. Or dinosaurs. This is no time to censor your inner kindergartner, go ahead and have fun. The best part is that you can always change it to something else later.

Found something you like? Good, save it to your Pictures folder.

Now go to your desktop and right click. A window pops up in Ubuntu 12.10 giving you the option to Change Desktop Background. Click that. In the new



window, click on the pull-down that says Wallpapers and select your Pictures folder.



Find the image you want and select it. Boom! You just customized your background. You can play with the pull down that says Zoom and see how that affects your wallpaper. Once you've found a look you like, click on the Behavior tab at the top of that window.

One of the fun things about Ubuntu is that you can go for a

really slick, uncluttered look. Under Behavior, you can tell Ubuntu to hide the icon strip on the left side of your screen when you're not using it. To do this, just switch the button next to Auto-Hide Launcher from Off to On. The Launcher disappears but is available at any time by simply moving your cursor all the way to the left side of your screen.

Neat, right?

Okay, close the window and take a look at your handiwork. Here's mine:



Aw, HELLS YEAH! My computer just locked the pod bay doors, Dave!

Amazing how just a tiny bit of tweaking can give your computer a whole new look and feel. In my case, the look and feel of a psychotic artificial intelligence.

Next, we're going to add some bling that will not only make your desktop look like it just stepped out of the Matrix, but will also give you useful information about what's going on under the hood.

One of the neatest little gems in the Ubuntu mine is a program called Conky (<http://conky.sourceforge.net/>). The description says it's a system monitor. But that's like saying Louboutins are shoes. There's so much more to it than that.

By tweaking Conky's configuration file (which is just a fancy way of saying, "Telling Conky what to do"), you can bring your desktop to glorious life.

First let's install Conky. Do it from the terminal so you maximize your Ubuntu bad-assery. Press

ASK THE NEW GUY

CTRL-ALT-T to open a terminal window, and type:

```
sudo apt-get install conky-all
```

Once you enter your password and Conky finishes installing, run it by typing:

conky

A display appears on your desktop, sitting over whatever wallpaper you chose. Below is what the default Conky setup looks like on my desktop.



Holy guacamole, I'm actually starting to fear my desktop.

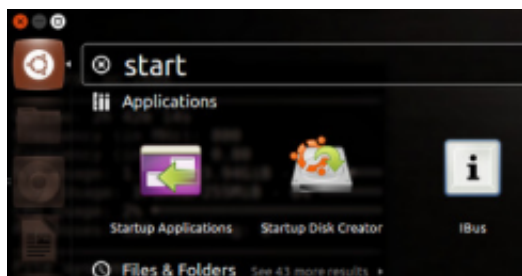
Awesome!

Conky gives you real-time updates about how much memory you're using, the programs you're running, how hard your computer

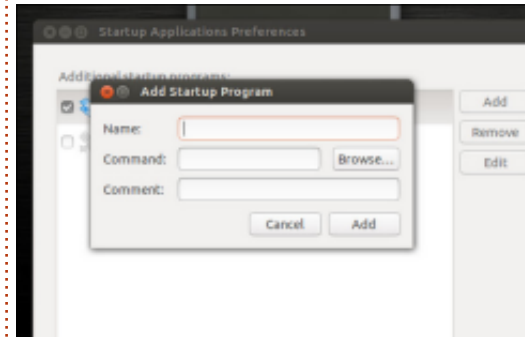
is working, etc. This is NOT the machine you take to the nursing home where you volunteer to help the residents check email. It's the machine you take to your high school bully's house to show him how hard you got over the years. If we did nothing else, we'd still have the most intimidating desktop on the block.

But remember that configuration file thingy I mentioned? You can tweak the settings that tell Conky what to display and how to display it. By changing the config file, you can get some of the incredible desktops shown in FCM: "My Desktop". Let's see if we can't have some fun.

The first thing we want to do is have Conky start whenever you boot up your computer. To do this, click on the Dash Home and type Start into the search field. You're looking for an Application called Startup Applications.



Click on that. In the window that comes up, click Add.



Where it says Name, type conky. Where it says Command, type conky. Yeah, I know this is complicated stuff; try to keep up.

Click Add and then close the window. Now, whenever you start your computer, Conky starts up as well.

Time to play with that config file. Start with a fresh new terminal window and a clean prompt by typing SHIFT-CTRL-T. At the prompt, type

```
cd /etc/conky/
```

We're telling Ubuntu to Change Directory to the Conky folder. Now type:

```
ls
```

You should see a list of files including one called conky.conf. In that file are all the commands that control how Conky looks and what it displays.

We're going to play with that file and see how it affects our Conky window. Before we do that, let's copy the file and create a backup in case we bork the whole thing and need to get back to our original setup. To do this, type:

```
sudo cp conky.conf
conkybackup.conf
```

We just told Ubuntu to Copy the file conky.conf and name the new file conkybackup.conf. Now we can tweak the original file but restore the defaults if we get in trouble (I'll show you how later).

Time to play. To open up the Conky configuration file, type

```
gksudo gedit conky.conf
```

Gedit is a standard text editor (you can use whatever you prefer), and we're telling it to open the conky.conf file. By using sudo, and entering your password when prompted, you are giving yourself permission to change the

conky.conf file.

Scroll through the conky.conf file. Any line that starts with a # is ignored by Conky. When you see something like 'alignment top_left' you can probably guess what that does. Let's live on the edge and change 'alignment top_left' to 'alignment top_right' and see what happens.

Click Save and close the gedit window. Back in your terminal, type:

```
conky
```

If you restart your computer at this point, the changes to the conky.conf file will take effect and you will see only one Conky window at the top right.

But don't restart yet. You can make even more radical changes to the conky.conf file – and then restart once you're happy with how everything looks.

Let's go back to playing. Get the conky.conf file open again by typing:

```
gksudo gedit conky.conf
```

What can we put in there? You can find some fun examples here (<http://ubuntuforums.org/showthread.php?t=281865&highlight=desktop+screenshot>). That thread has over TWO THOUSAND PAGES of custom Conky config files you can use!

I picked the first one, copied the code and pasted it over the text in my conky.conf file. You can do the same. From the following link, copy and paste the lines of code into your conky.conf file (make sure you get all of it and completely replace what was already in the file): <http://pastebin.com/Bda2V62Q>

Click save and close gedit.

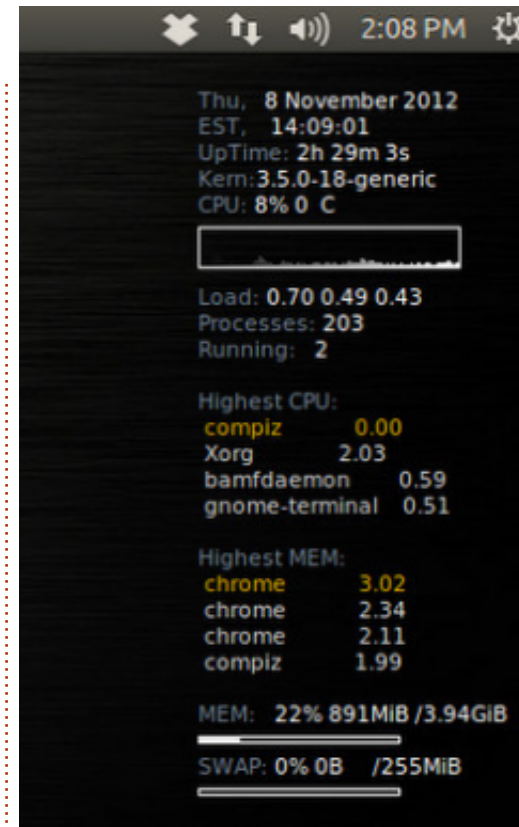
Open a new Terminal window and type:

```
killall conky
```

Then type:

```
conky
```

Killall terminates the original conky session so you can run it again with the new setup. If everything went to plan, you should see something like this



Now how cool is that?! The correct answer is ALL. All cool.

This is just the beginning! There are ways to tell Conky to display weather, use pretty graphics, display data horizontally, etc. You can even call up multiple Conky windows and place them in different areas on your screen! Go grapenuts!

There's a TON of resources for making Conky look incredible but

also give you useful updates. Start with this Ubuntu community page (<https://help.ubuntu.com/community/SettingUpConky>) and then look around for interesting desktops.

By the way, you'll see references to a .conkyrc file. Creating a .conkyrc file works just as well as what I did here. I prefer backing up the conky.conf file and then tweaking it directly just because it keeps me in the same directory as my backup in case I mess up. I'm less likely to get confused that way. But do whatever works for you.

Have fun making your computer look like Hal from 2001! But just, you know, hang on to a spare set of keys for the pod bay.



Copil is an Aztec name that roughly translates to "you need my heart for what again?" His love of women's shoes is chronicled at yaconfidential.blogspot.com. You can also watch him embarrass himself on Twitter (@copil).



For years, we've had a bit of a giggle, poking fun at Microsoft whenever they've done something even remotely bad, like collecting usage data. With the release of Ubuntu 12.10, it's Canonical's turn to do something shockingly bad, by sending your keystrokes to Facebook, Amazon, and other partners. It's a feature that's turned on by default. Even worse, there's no consent window warning you Canonical is going to record your Unity dash search keystrokes, only a tiny info button at the bottom-right of the dash which leads to a quite scary looking notice (right).

This "feature" can be turned off via the Privacy panel in System Settings, but it's on by default and not very obvious unless you happen to click on that small info button.

If you can get past that, there's a lot going on in this new release.

I tested Ubuntu 12.10 on an HP Compaq 6710b notebook which

has a Core 2 Duo T7100 1.8 GHz CPU (2 cores), 2 GB RAM, Intel GM965 graphics (256 MB), and an 80 GB hard drive.

The installation went off without a hitch, and no work was required to get the wireless working. The only feature on the notebook which didn't work out-of-the-box was the integrated fingerprint reader. Unity performance is good with 2GB of

RAM. It feels a bit snappier than 12.04 did, and I rarely get application crashes, though they do happen sometimes.

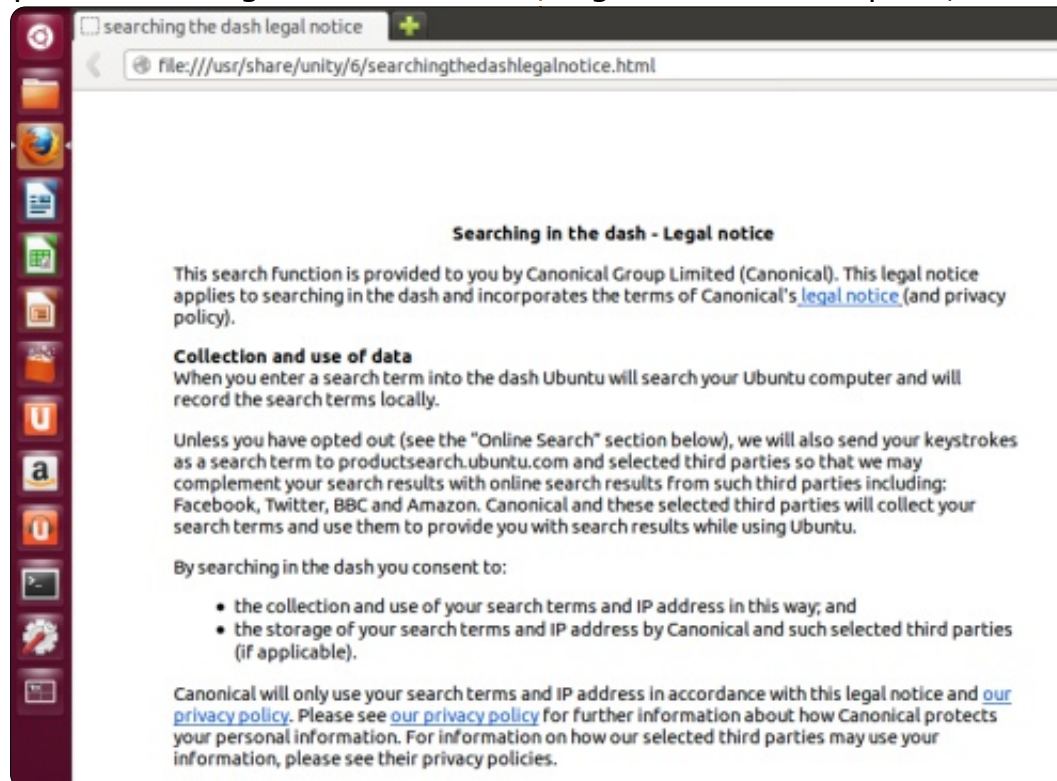
The first obvious change is the addition of the ability to remotely log in to another machine via RDP, directly from the login screen. This means you don't actually have to be logged in to the computer sitting in front of you to remotely log into another computer, a handy

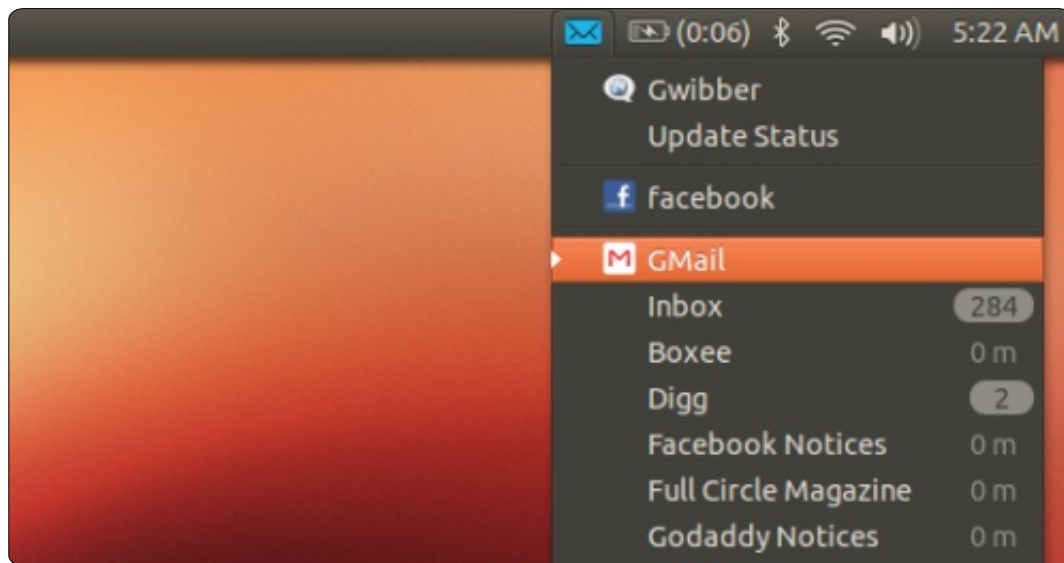
feature in mixed networks where you have to log into Windows machines.

The next obvious change is the addition of the Amazon store to the Unity panel, which lead me to finding out about Unity's new web application integration. Popular websites like Gmail, Facebook, and OMGUbuntu are better integrated into Unity: a pop-up appears asking if you want to install. Integration adds features like being able to access your messages from the mail drop down at the top of the screen.

Social networks are also integrated better in Ubuntu 12.10. Not only do networks appear in the top panel, but they're searchable through the Unity lens, if you have the network configured. This is a really handy feature when you're trying to remember who tweeted a particular message hundreds of tweets ago. I tested searching identi.ca messages.

Perhaps the coolest feature



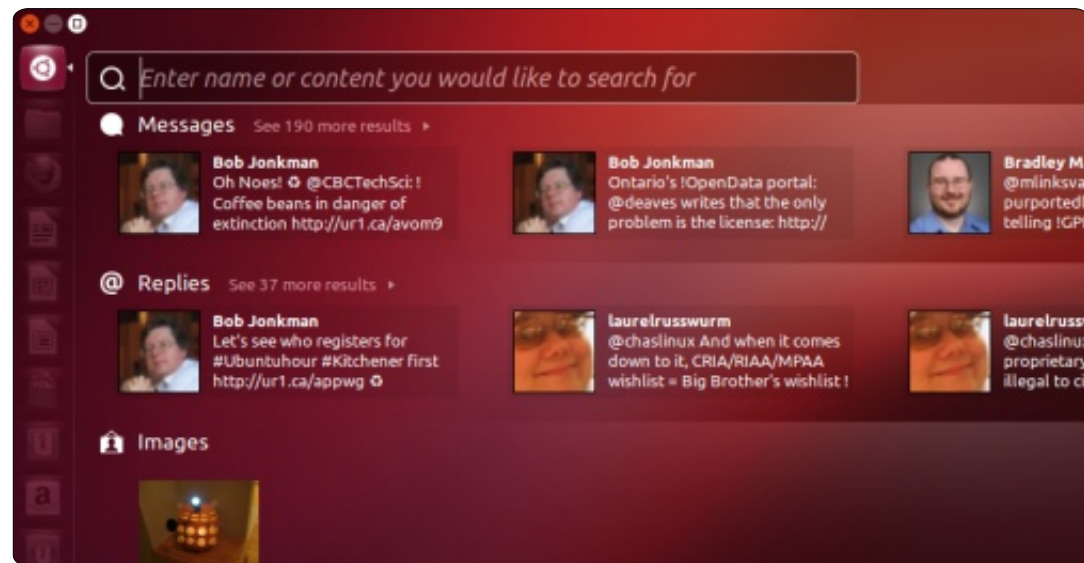
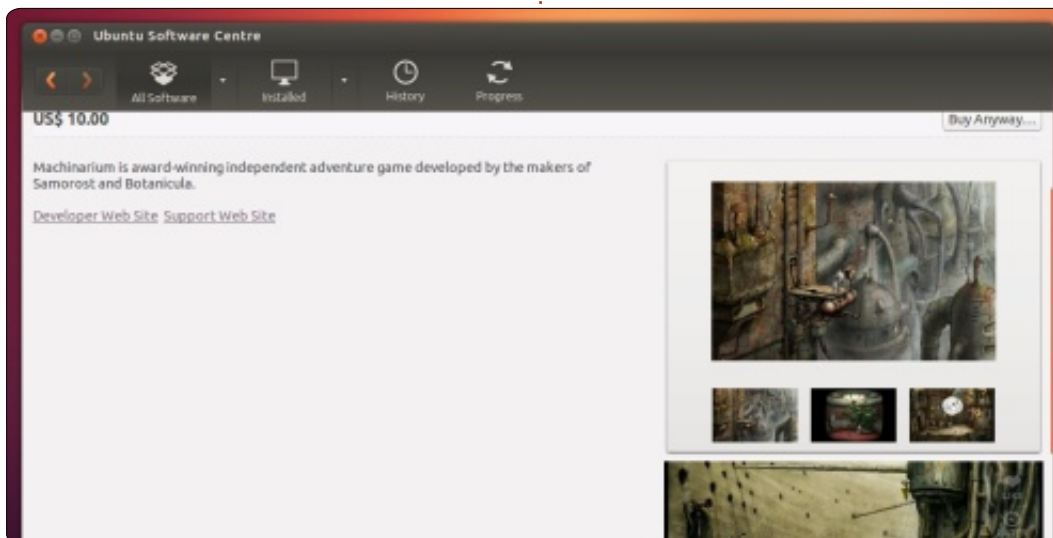


that's been added is the ability to preview certain content in Unity, music for example. Right clicking on search results brings up information about the music, and sometimes a preview of the music. Occasionally, parts of entire albums can be previewed before paying to download the music.

Preview also works for movies, pictures, and documents, though previews for movies actually open up Firefox and run on Youtube. Picture preview allows you to email the previewed picture, but here's where some of that web application integration falls down. Picture/Document attachments default to Thunderbird (which I didn't have set up) rather than

Google Mail.

The Ubuntu Software Centre has video previews for some programs. The video previews I checked out were all through Vimeo, a strange mix using Youtube in the Unity lens and



Vimeo in USC. Software Centre video previews can be expanded to full-screen. I spoke with friends using Ubuntu 12.04 and Ubuntu 12.10, and the few who have been buying apps said they haven't had any problems so far.

Ubuntu 12.10 is an improvement from a performance standpoint. Canonical has introduced a few new features, but the features need work before they are really useful. And, for some of those features, sending keystrokes to Amazon/Facebook/etc, Canonical should get a slap on the wrist.



Charles is a step-father, husband, and Linux fan, who hosts a not-for-profit computer refurbishing project. When not breaking hardware/servers, he maintains a blog at:
<http://www.charlesmccolm.com/>



CLOSING WINDOWS

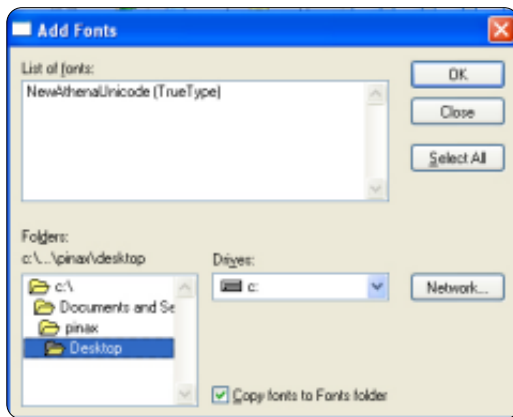
Written by:

Ronnie Tucker (KDE)

Jan Mussche (Gnome)

Elizabeth Krumbach (XFCE)

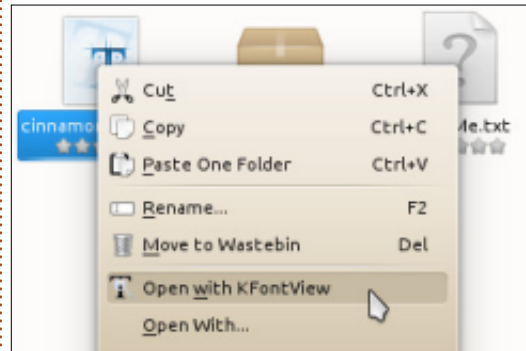
Mark Boyajian (LXDE)



Every now and then, you'll want to add some pizzazz to your Word document or presentation. This is usually done by adding fonts to your Windows XP system. It's as simple as giving XP a folder of fonts (bottom left in the above image) and clicking OK. The fonts will be added to your system and be available to any application the next time the app is loaded.

Kubuntu

Kubuntu (and KDE in general) makes it even easier to add a font. Simply right-click any Windows compatible font file and choose to open it with KFontView:



This will give you a preview of the font, and, if you wish to install the font, simply click the 'Install...' button at the bottom right of the window.

Gnome-Shell

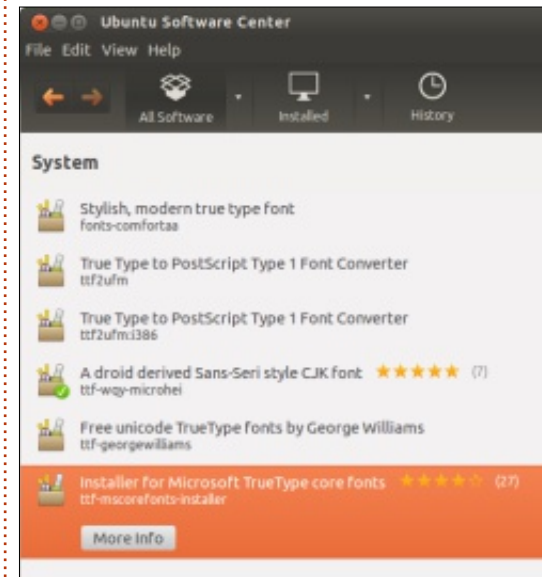
When using Gnome, there is a semi-automatic way of installing extra fonts. Simply use the Software center. Type font or True Type Font in the search field, and you'll get a list of all fonts available in your system. Choose the one you want to install and let the program do the rest.

Another way is to download font files and store them in your home directory. Now double-click the file (with extension fon, fnt, ttc, ttf or otf), and you see examples of the lettertype. At the bottom-right, you can click on the button Install Font. If everything could be that simple!

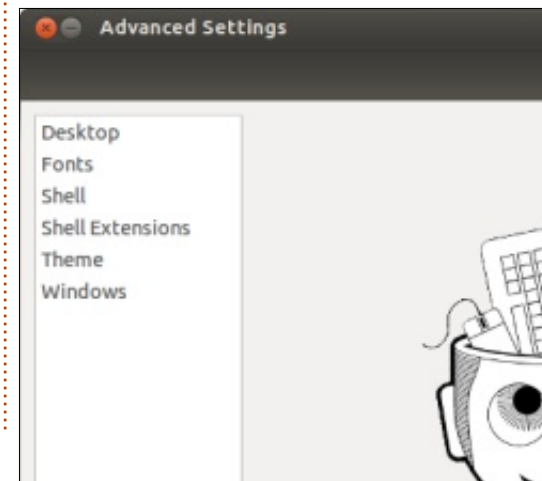
With gnome-tweak-tool, it is possible to change the default font for certain items. Open the program and click in the left

Adding Fonts

column on Fonts. Now it is possible



to change the default font, monospace, document font, and window title font. Also you can set the values for hinting and antialiasing.



Xubuntu

Given the minimalist approach of the file manager in Xfce, there is no automatic way of installing fonts in Xubuntu. However, it only takes a few steps to install a font manually.

To install a font on your whole system, you can do the following:

Open the file manager (thunar) as an administrative user with this command:

```
gksudo thunar
```

Navigate to `/usr/share/fonts/`, create a new folder for your fonts, and move your `.ttf` files to it.

Close thunar and run this command for the system to pick up the changes:

```
sudo fc-cache -f
```

Alternatively, if you wish to install (or restrict access to) the font for only your own use, you can do the following:

Create a `.fonts/` directory in your home directory and move

your `.ttf` files to it.

Then, for the system to pick up the changes, just run:

```
fc-cache -f
```

Now your new fonts should be available to your applications and desktop environment.

Lubuntu

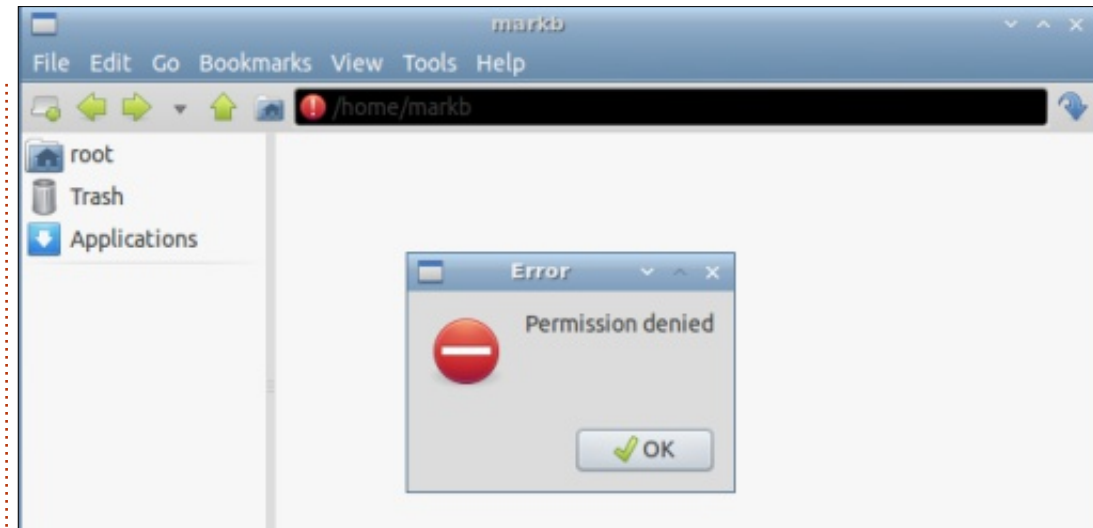
As with Xubuntu above, there is no “automatic” processing to install fonts in Lubuntu. In fact, the process in Lubuntu is virtually the same as in Xubuntu, so there seems little point to reiterate what has already been competently documented [under Xubuntu]. Two quick points of distinction, however, for Lubuntu:

In Lubuntu, the file manager is PCManFM. You can open PCManFM as superuser in one of two ways:

From the command line, enter:

```
gksudo pcmanfm
```

You are then prompted for your root password. Upon providing your password and clicking OK (or pressing Enter), PCManFM will



open – along with a small dialog box which states: Error Permission Denied.

This is extremely misleading because the message is wrong! You do, in fact, have permission and PCManFM is, in fact, running as root, so treat it with respect! Simply click OK in the dialog box to close it, and you are ready to navigate your file system as root using PCManFM.

One important caveat: you will notice that the path field is “blacked out” with a red “bang” at the left end of the field. To be honest, I have no idea what the actual intent of this behavior is supposed to accomplish other than to remind you that you are running

as root (and that’s a very important visual aid!); however, I discovered that triple-clicking in the path field will reveal the path string. In fact, once you’ve triple-clicked in this field, it can be used as in a “normal” (i.e. non-root) instance of PCManFM; however, each time the path changes, you need to triple-click in the field again to reveal its contents. If all this is intended to constantly remind you that you are running as root, then that’s fantastic; otherwise, it can be a bit of nuisance.

You can, of course, open PCManFM as a user from the GUI (Accessories > File Manager), and then invoke a root instance. Once PCManFM is open, you can navigate to any folder, then, from

the menu bar, select Tools > Open Current Folder as Root. You will then be prompted for your root password, and a new instance of PCManFM will open as root. Unlike opening as root from the command line, you are not provided with the misleading dialog box; however, all other functionality is the same.

NOTE: Once you have a root instance open, you will have root privileges to any folder/file to which you navigate! Root privileges are not limited only to the folder from which you invoked the root instance of file manager, even though that seems to be implied by the menu title ("Open Current Folder as Root"). Be warned!!

You are not limited to installing only .ttf font files in the fonts directory (either the system directory or the user directory); you can also install .otf (OpenType Font) font files and they will work, too.

A quick comment on fonts and font management in minimalist distros like Lubuntu: There is a wonderful font management program available in most

repositories called FontyPython. This lovely little program excels at managing your installed fonts, displays a preview of selected fonts, and allows you to create "Pogs": custom collections of fonts. And, as if that's not enough, the "Monty Python" parallels are carried throughout the application; quite humorous.

So, even though Lubuntu doesn't come with "fancy" font previewing software preinstalled, it's a snap to drop it in and get even more nifty functionality (like Pogs), not to mention clever "Python-isms".

QUICK MY STORY

In 2007 my husband bought me a laptop as a surprise. It was an Acer Aspire loaded with Vista. I struggled with it for about six weeks and got more and more frustrated facing 'Allow or Deny?' at almost every step.

I had been using computers for quite a long time at that point, starting with an Apple and Win3.1 back in the early 90s, working up to an iMac at home and Windows at the university where I was working.

I'd been a member of an online forum where I became friends with the tech guy of a company in the mid-west. He'd been talking to me for a while about Linux, in his personal life and also on the company computers. So much of what he said was intriguing that I started to think that maybe I could try Linux, too.

One October weekend in particular I'd had enough. I started looking through various websites trying to get a sense of what flavor of Linux was the easiest to use for newbies, had the best support as far as technical information and moral support for those with no previous experience. Everything pointed to Ubuntu. I downloaded 7.04 and dumped Vista.

My Acer was equipped with wireless, but it wasn't working. I started investigating that and learned more about my system in that time than I had previously learned about PCs. Then the "Update" dialog popped up and I was notified that 7.10 was available. I downloaded all the updates and restarted. The system came back on, and I just happened to be staring at the indicator area when the wired icon went out and was replaced with a wireless icon! I was hooked right there, and was a happy convert after that.

Amanda Makepeace



MY STORY

This story begins at the beginning of 2010. I was broke at the time, so I was trying to find a free operating system. I needed something that could run on my PCs at home. I had searched on the Internet, but found nothing useful for a long time. But one day I was at Barnes and Noble, and I saw a magazine for Linux. Although I had heard of Linux before, I never

thought of it as something I would ever be able to use. When I asked people whom I knew were computer professionals, I was told it was for people who were experts, and it was difficult to use. I never heard anything positive about it. I am so amazed that I hadn't come across it sooner.

When I read the magazine, I became exposed to Ubuntu 9.10,

Karmic Koala. It sounded so good, as if it was exactly what I was looking for. As a result, I got very excited, took it home, and to my surprise had such an easy time installing it to my PC that I decided to run it along with Windows XP as a dual boot system. All I did was put the live CD in the drive and the instructions were step-by-step—you would have to be pretty slow to not understand how to set things up.

Since then, I have been very satisfied with Ubuntu in general, and I have been able to check out later versions of it from 10.04 to 12.04.

Anthony Venable

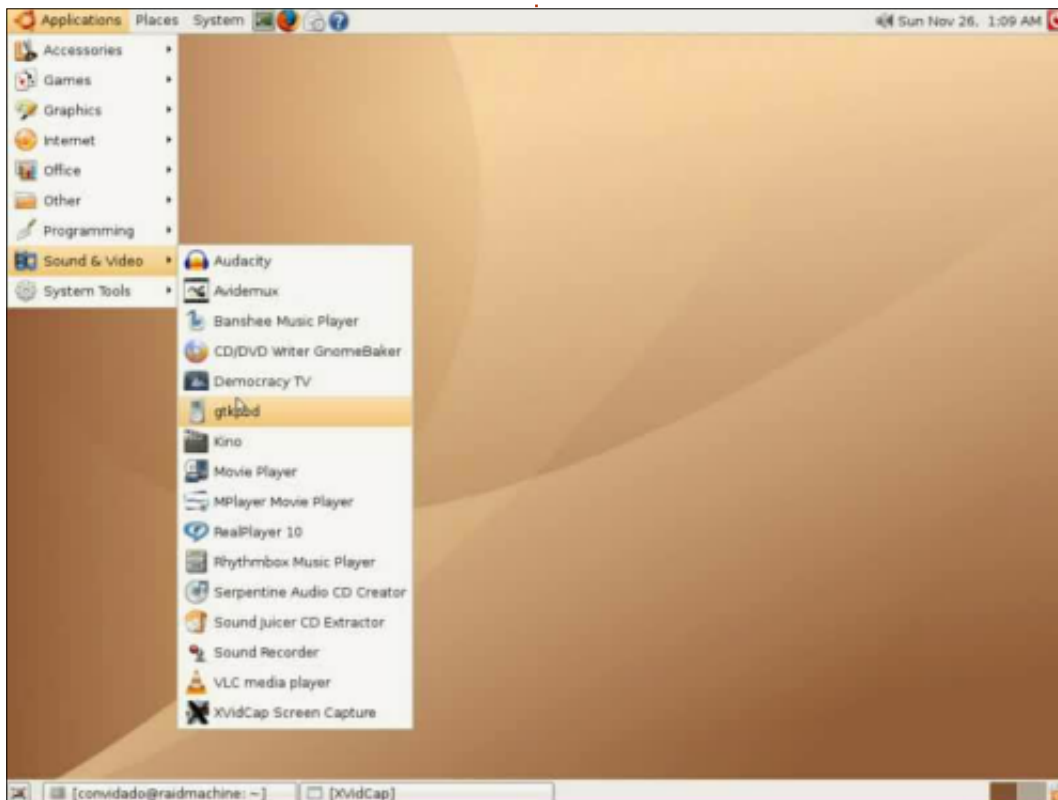
The day I got my brand new Compaq laptop from my local tech shopping mall, I bought some blank CDs to do some burning of the Ubuntu OS. I had also done a little bit of reading about installing Ubuntu before that. It was, errrr, a

nightmare revisited!

Firstly, my friend Faizal recommended that I install the Netbook version 10.10—without knowing that it should be for netbook, not laptop! I went through the installation process, and... BAMB! Display resolution all wrong; no sound; no nothing! I was, like, OMG, here I go again!

When I related my terrible nightmare to Faizal, he said it could be a version problem. Netbooks have limited support in terms of hardware. I was, like, ok, yeah, that makes sense to me. I proceeded to install the Desktop version 10.10, and, BAMB again—another display resolution all wrong, still no sound, I got the X Server bootup problem but it wouldn't auto boot to GUI.

I combed through the Ubuntu support forum online, looking for answers and solutions. Perhaps someone out there has done it before, and has the answers I am looking for? It had been the 5th day since I started my venture with Ubuntu. Sleepless nights—digging



the internet and the Ubuntu Support Forum for an answer. No luck in my editing of configuration files, driver updates, or new packages installation.

I posted a help call in Ubuntu FB twice, and suddenly some Chinese fellow told me to re-compile the Kernel to an older version. A bit of a blur, I Google up the Kernel version. I got the answer that it's an older version of Ubuntu version 10.04. I reposted my question to the Chinese fellow and he told me that it's the only way to get my laptop running and speaking to me. Well, if that's the solution to end my sleepless nights and nightmare, let it be.

I got Ubuntu 10.04 LTS, and ready for a final showdown! It has to work this time or I might go MIA again! After the installation ended, I rebooted my laptop and waited a while as I searched for more possible answers online with my old laptop. It booted straight into GUI with beautiful colors but no sound coming from my Altec Lansing. I tried plugging in the sound port, and I could hear soft music. I knew my laptop was trying to speak to me. Back to combing through Ubuntu Support forum

again. I got a suggestion to upgrade the ALSA and Realtek drivers to the latest versions.

I rebooted and then a sound came from my new laptop! It was calling, talking to me! I was, like, OMG! It was the most beautiful sound coming from my new Altec Lansing laptop since I had bought it a week previously. As it booted up, it continued to go straight into the GUI. Then I double-checked the display resolution, webcam, USB connection, WIFI, and it was ALL

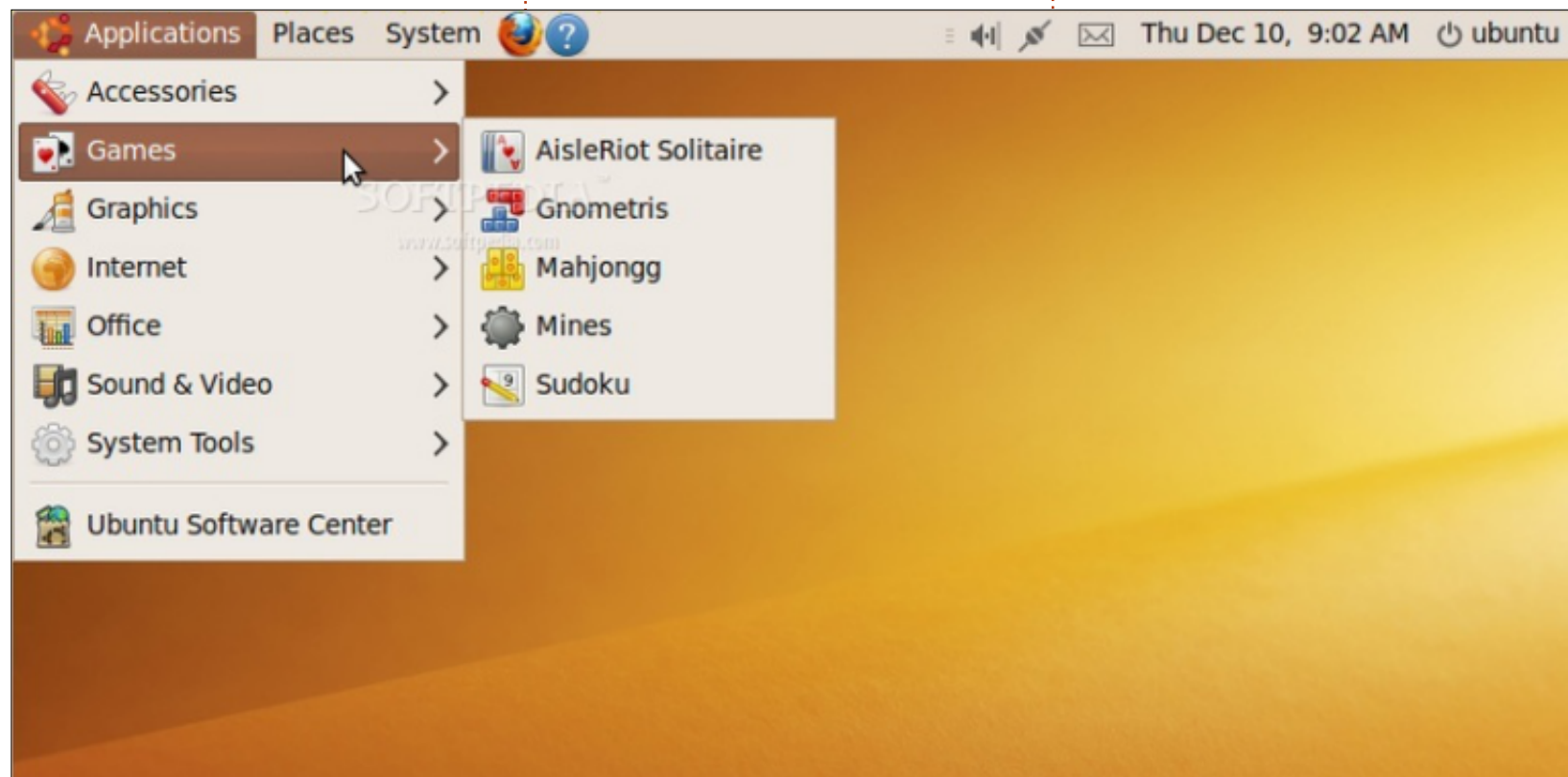
PERFECT! As I started to venture around the Ubuntu system, I finally felt at peace! It was already on the 7th day that I was able to sleep a silent night. I felt happiness and joy that my journey of the re-entry has been completed.

After a few more days of toying and playing with the system, I got my Ubuntu to look like a Mac OSX! How cool is that? WAY COOL I say! I managed to post on Twitter, FB, chats, emails, played a few simple small games, and it all worked just

fine. All the while, the system never hung or froze like my old laptop, which was pretty annoying. Linux, it's so good to see you again my old friend. There you are. I am glad that I haven't lost all of my Linux touch yet!

I have now upgraded to 12.04. Some bugs in Gnome are still fighting inside with Unity, and it's a work-in-process to find a fix now.

Margaret Chong





MY OPINION

Written by Irv Risch

PCMAG.COM had an article entitled "Clearing Up 5 Windows 8 Confusions" that made me do some thinking:
<http://www.pcmag.com/article2/0,2817,2411466,00.asp>. Most people out there use computers (desktop or laptops), and carry either a smartphone or tablet or some portable device. Now, we are using different operating systems on these devices. Some of the most popular ones are Android, iOS, and Windows. Besides the OS, we also use different desktops.

This article talked about confusions in Windows 8, but we also have confusions about other things that we use. When we think about Tablet OS or PC OS, there is a difference here as well. For example, if you are using Android on a tablet: it works great with a touch-screen, but, when you use a mouse and keyboard, you run into problems. Just try the navigation keys and shortcut keys, and you will see what I mean.

Now, in Windows 8, you find the

new style (formerly known as Metro) vs. desktop. One is designed to work with touch-screens, and the other to work with PCs with a desktop. The same is true with an OS designed for things like Android Touch Screen devices. Remember, Android is Linux-based. Thus it is open-source and there are many versions of Android. For example, my wife has a Chromebook with Google OS—which is Android. But this does not have a touch-screen, so Android can be designed to work with devices without touch-screens. At this point, I need to mention that, if you have a Nexus 7, you are running Google Android that is designed for touch-screen devices.

So, what is the difference between Windows 8 and Windows RT? Keep this in mind when you think of Win RT; it is purely a tablet version of Windows 8. This is no different than my Android (Linux) running on my B&N Nook, which is designed for a touch-screen device.

Now, with all this talk about

touch-screen devices with their OS, I must mention that Apps need to be developed to run on these devices, e.g., Angry Birds. You can find this app on the B&N online store for the Nook, Amazon store for the Kindle and many other devices, and Google store for the Nexus 7. And now we have a Windows 8 store with apps for Win RT devices. Remember this fact, that these apps are developed for all these touch-screen devices, and not for desktops and laptops.

If you are using the Google Chrome Browser, you can run some of the same apps that run on Google Chromebook. I know this for a fact because I do it myself. I am not a developer, but I understand that there is a difference in the design of these apps from programs that run on computers. The way things work today are different than a few years ago. Previously, you turned your computer on and your OS loaded. When you were at your desktop, you started your word-processor and started typing. Everything was on your computer.

Now, you turn on your computer, your OS loads. You go to your Web Browser, click on an app like Google Docs, and start typing. Everything is in the cloud. This is the future.

All this time that I have been writing about Windows 8, Android, etc, I've had Ubuntu Linux in the back of my mind. Right now, things are like this: I have my laptop running Ubuntu, and my tablet running Android. I don't run my Android apps on my laptop, and I don't run my laptop programs on my tablet. But I am looking forward to the day when I can have a machine that runs the same applications. So, no matter whether I use it on a laptop or tablet, I will be on the same page. Yes, Windows 8 is designed to do this, but, remember I said you have Windows 8 and you have Windows RT. However, with Ubuntu with the Unity desktop, you will be able to run it on your tablet, TV, Phone, Computer, and other devices. And, as far as I know, you will be able to run the same programs. This is the way it should be.





WHAT IS...

Written by Wikipedia

NEW

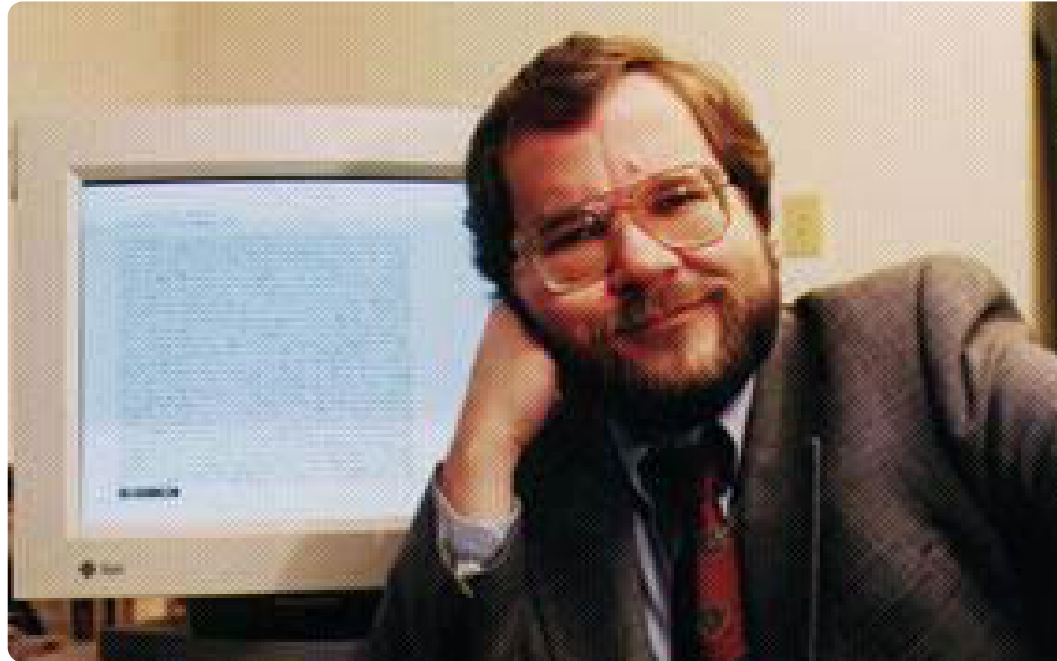
PGP Encryption

I hope this will become a new column. I'd like to use this space to show articles that take a single subject and explain (briefly, and simply) the history, some behind-the-scenes information, and how the software we use interacts with the chosen subject. Send your articles to

articles@fullcirclemagazine.org

Here are some ideas to help get you creative: *newsgroups, IRC, BitTorrent, email, the web, video codecs, the kernel, graphics tablets...*

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting and decrypting texts, e-mails, files, directories and whole disk partitions to increase the security of e-mail communications. It was created by Phil Zimmermann in



1991.

PGP and similar products follow the OpenPGP standard (RFC 4880) for encrypting and decrypting data.

Design

PGP encryption uses a serial combination of hashing, data compression, symmetric-key cryptography, and, finally, public-key cryptography; each step uses one of several supported

algorithms. Each public key is bound to a user name and/or an e-mail address. The first version of this system was generally known as a web of trust to contrast with the X.509 system, which uses a hierarchical approach based on certificate authority and which was added to PGP implementations later. Current versions of PGP encryption include both options through an automated key management server.

Compatibility

As PGP evolves, PGP that support newer features and algorithms are able to create encrypted messages that older PGP systems cannot decrypt, even with a valid private key. Thus, it is essential that partners in PGP communication understand each other's capabilities or at least agree on PGP settings.

Confidentiality

PGP can be used to send messages confidentially. For this, PGP combines symmetric-key encryption and public-key encryption. The message is encrypted using a symmetric encryption algorithm, which requires a symmetric key. Each symmetric key is used only once and is also called a session key. The session key is protected by encrypting it with the receiver's public key thus ensuring that only the receiver can decrypt the session key. The encrypted message along with the encrypted



WHAT IS

session key is sent to the receiver.

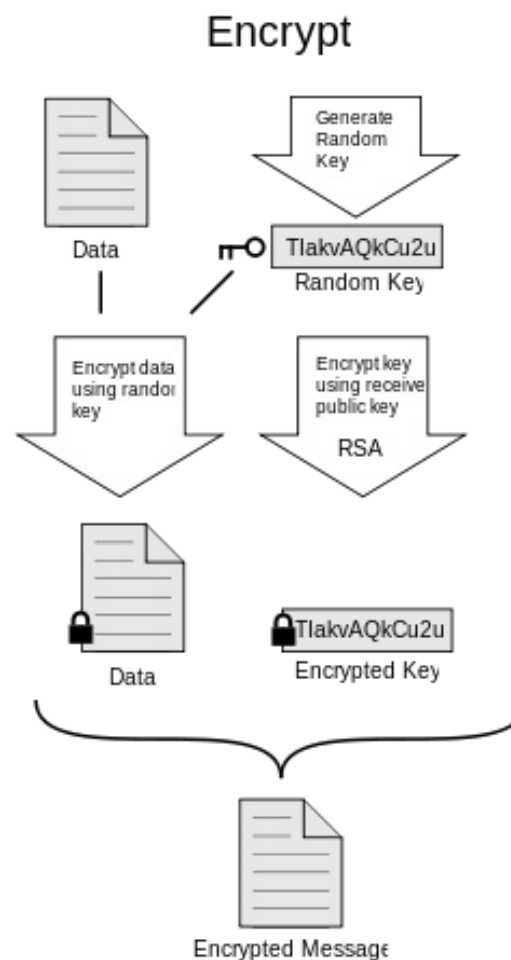
Digital signatures

PGP supports message authentication and integrity checking. The latter is used to detect whether a message has been altered since it was completed (the message integrity property), and the former to determine whether it was actually sent by the person/entity claimed to be the sender (a digital signature). Because the content is encrypted, any changes in the message will result in failure of the decryption with the appropriate key. The sender uses PGP to create a digital signature for the message with either the RSA or DSA signature algorithms. To do so, PGP computes a hash (also called a message digest) from the plaintext, and then creates the digital signature from that hash using the sender's private key.

Web of trust

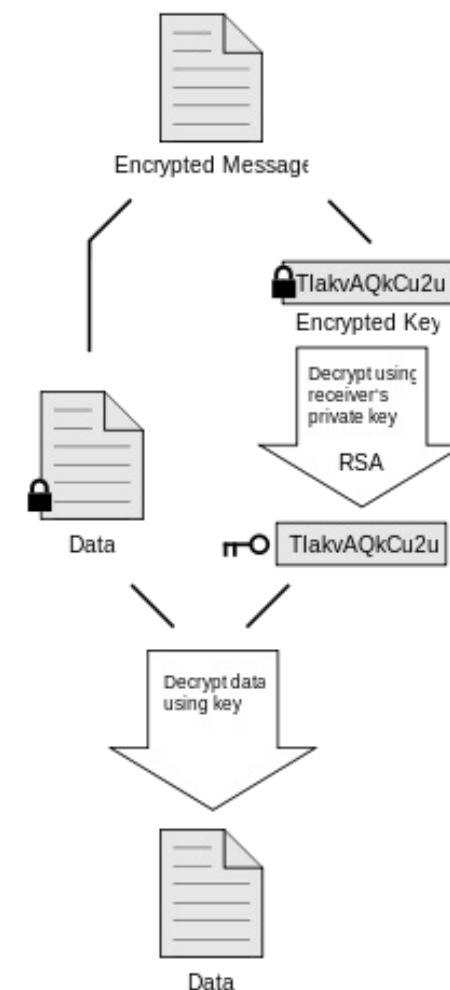
Both when encrypting messages and when verifying signatures, it is critical that the public key used to send messages to someone or some entity

actually does 'belong' to the intended recipient. Simply downloading a public key from somewhere is not overwhelming assurance of that association; deliberate (or accidental) impersonation is possible. PGP has, from its first versions, always included provisions for distributing a user's public keys in an 'identity certificate' which is also constructed cryptographically so that any tampering (or accidental garble) is readily detectable. But merely making a certificate which is impossible to modify without being detected effectively is also insufficient. It can prevent corruption only after the certificate has been created, not before. Users must also ensure by some means that the public key in a certificate actually does belong to the person/entity claiming it. From its first release, PGP products have included an internal certificate 'vetting scheme' to assist with this; a trust model which has been called a web of trust. A given public key (or more specifically, information binding a user name to a key) may be digitally signed by a third party user to attest to the association between someone (actually a user name) and the key. There are



several levels of confidence which can be included in such signatures. Although many programs read and write this information, few (if any) include this level of certification when calculating whether to trust a key.

Decrypt



The web of trust protocol was first described by Zimmermann in 1992 in the manual for PGP version 2.0:

As time goes on, you will accumulate keys from other

people that you may want to designate as trusted introducers. Everyone else will each choose their own trusted introducers. And everyone will gradually accumulate and distribute with their key a collection of certifying signatures from other people, with the expectation that anyone receiving it will trust at least one or two of the signatures. This will cause the emergence of a decentralized fault-tolerant web of confidence for all public keys.

The web of trust mechanism has advantages over a centrally managed public key infrastructure scheme such as that used by S/MIME but has not been universally used. Users have been willing to accept certificates and check their validity manually or to simply accept them. No satisfactory solution has been found for the underlying problem.

Certificates

In the (more recent) OpenPGP specification, trust signatures can be used to support creation of certificate authorities. A trust signature indicates both that the key belongs to its claimed owner,

and that the owner of the key is trustworthy to sign other keys at one level below their own. A level 0 signature is comparable to a web of trust signature since only the validity of the key is certified. A level 1 signature is similar to the trust one has in a certificate authority because a key signed to level 1 is able to issue an unlimited number of level 0 signatures. A level 2 signature is highly analogous to the trust assumption users must rely on whenever they use the default certificate authority list (like those included in web browsers); it allows the owner of the key to make other keys certificate authorities.

PGP versions have always included a way to cancel ('revoke') identity certificates. A lost or compromised private key will require this if communication security is to be retained by that user. This is, more or less, equivalent to the certificate revocation lists of centralized PKI schemes. Recent PGP versions have also supported certificate expiration dates.

The problem of correctly identifying a public key as belonging to a particular user is

not unique to PGP. All public key / private key cryptosystems have the same problem, if in slightly different guise, and no fully satisfactory solution is known. PGP's original scheme, at least, leaves the decision whether or not to use its endorsement/vetting system to the user, while most other PKI schemes do not, requiring instead that every certificate attested to by a central certificate authority be accepted as correct.

Security quality

To the best of publicly available information, there is no known method which will allow a person or group to break PGP encryption by cryptographic or computational means. Indeed, in 1996, cryptographer Bruce Schneier characterized an early version as being "the closest you're likely to get to military-grade encryption." Early versions of PGP have been found to have theoretical vulnerabilities and so current versions are recommended. In addition to protecting data in transit over a network, PGP encryption can also be used to protect data in long-term data

storage such as disk files. These long-term storage options are also known as data at rest, i.e. data stored, not in transit.

The cryptographic security of PGP encryption depends on the assumption that the algorithms used are unbreakable by direct cryptanalysis with current equipment and techniques. For instance, in the original version, the RSA algorithm was used to encrypt session keys; RSA's security depends upon the one-way function nature of mathematical integer factoring. Likewise, the symmetric key algorithm used in PGP version 2 was IDEA, which might, at some future time, be found to have a previously unsuspected cryptanalytic flaw. Specific instances of current PGP, or IDEA, insecurities—if they exist—are not publicly known. As current versions of PGP have added additional encryption algorithms, the degree of their cryptographic vulnerability varies with the algorithm used. In practice, each of the algorithms in current use is not publicly known to have cryptanalytic weaknesses.

New versions of PGP are released periodically and

vulnerabilities are fixed by developers as they come to light. Any agency wanting to read PGP messages would probably use easier means than standard cryptanalysis, e.g. rubber-hose cryptanalysis or black-bag cryptanalysis i.e. installing some form of trojan horse or keystroke logging software/hardware on the target computer to capture encrypted keyrings and their passwords. The FBI has already used this attack against PGP in its investigations. However, any such vulnerabilities apply not just to PGP, but to all encryption software.

In 2003, an incident involving seized Psion PDAs belonging to members of the Red Brigade indicated that neither the Italian police nor the FBI were able to decrypt PGP-encrypted files stored on them.

A more recent incident in December 2006 (see *United States v. Boucher*) involving US customs agents and a seized laptop PC which allegedly contained child pornography indicates that US Government agencies find it "nearly impossible" to access PGP-encrypted files. Additionally, a

judge ruling on the same case in November 2007 has stated that forcing the suspect to reveal his PGP passphrase would violate his Fifth Amendment rights i.e. a suspect's constitutional right not to incriminate himself. The Fifth Amendment issue has been opened again as the case was appealed and the federal judge again ordered the defendant to provide the key.

Evidence suggests that as of 2007, British police investigators are unable to break PGP, so instead have resorted to using RIPA legislation to demand the passwords/keys. In November 2009, a British citizen was convicted under RIPA legislation and jailed for 9 months for refusing to provide police investigators with encryption keys to PGP-encrypted files.

History

Early history

Phil Zimmermann created the first version of PGP encryption in 1991. The name, "Pretty Good Privacy", is humorously ironic and was inspired by the name of a

grocery store, "Ralph's Pretty Good Grocery", featured in radio host Garrison Keillor's fictional town, Lake Wobegon. This first version included a symmetric-key algorithm that Zimmermann had designed himself, named BassOmatic after a Saturday Night Live sketch. Zimmermann had been a long-time anti-nuclear activist, and created PGP encryption so that similarly inclined people might securely use BBSs and securely store messages and files. No license was required for its non-commercial use. There was not even a nominal charge, and the complete source code was included with all copies.

In a posting of June 5, 2001, entitled "PGP Marks 10th Anniversary", Zimmermann describes the circumstances surrounding his release of PGP:

"It was on this day in 1991 that I sent the first release of PGP to a couple of my friends for uploading to the Internet. First, I sent it to Allan Hoeltje, who posted it to Peacenet, an ISP that specialized in grassroots political organizations, mainly in the peace movement. Peacenet was accessible to political activists all over the world.

Then, I uploaded it to Kelly Goen, who proceeded to upload it to a Usenet newsgroup that specialized in distributing source code. At my request, he marked the Usenet posting as "US only". Kelly also uploaded it to many BBS systems around the country. I don't recall if the postings to the Internet began on June 5th or 6th.

It may be surprising to some that back in 1991, I did not yet know enough about Usenet newsgroups to realize that a "US only" tag was merely an advisory tag that had little real effect on how Usenet propagated newsgroup postings. I thought it actually controlled how Usenet routed the posting. But back then, I had no clue how to post anything on a newsgroup, and didn't even have a clear idea what a newsgroup was."

PGP found its way onto the Internet, and it very rapidly acquired a considerable following around the world. Users and supporters included dissidents in totalitarian countries (some affecting letters to Zimmermann have been published, some of which have been included in testimony before the US

Congress), civil libertarians in other parts of the world (see Zimmermann's published testimony in various hearings), and the 'free communications' activists who called themselves cypherpunks (who provided both publicity and distribution), and, decades later, CryptoParty, who did much the same via Twitter.

Criminal investigation

Shortly after its release, PGP encryption found its way outside the United States, and, in February 1993, Zimmermann became the formal target of a criminal investigation by the US Government for "munitions export without a license". Cryptosystems using keys larger than 40 bits were then considered munitions within the definition of the US export regulations; PGP has never used keys smaller than 128 bits, so it qualified at that time. Penalties for violation, if found guilty, were substantial. After several years, the investigation of Zimmermann was closed without filing criminal charges against him or anyone else.

Zimmermann challenged these

regulations in a curious way. He published the entire source code of PGP in a hardback book, via MIT Press, which was distributed and sold widely. Anybody wishing to build their own copy of PGP could buy the \$60 book, cut off the covers, separate the pages, and scan them using an OCR program, creating a set of source code text files. One could then build the application using the freely available GNU Compiler Collection. PGP would thus be available anywhere in the world. The claimed principle was simple: export of munitions—guns, bombs, planes, and software—was (and remains) restricted; but the export of books is protected by the First Amendment. The question was never tested in court with respect to PGP. In cases addressing other encryption software, however, two federal appeals courts have established the rule that cryptographic software source code is speech protected by the First Amendment (the Ninth Circuit Court of Appeals in the Bernstein case and the Sixth Circuit Court of Appeals in the Junger case).

US export regulations regarding cryptography remain in force, but were liberalized substantially

throughout the late 1990s. Since 2000, compliance with the regulations is also much easier. PGP encryption no longer meets the definition of a non-exportable weapon, and can be exported internationally except to 7 specific countries and a list of named groups and individuals (with whom substantially all US trade is prohibited under various US export controls).

PGP 3 and founding of PGP Inc.

During this turmoil, Zimmermann's team worked on a new version of PGP encryption called PGP 3. This new version was to have considerable security improvements, including a new certificate structure which fixed small security flaws in the PGP 2.x certificates, as well as permitting a certificate to include separate keys for signing and encryption. Furthermore, the experience with patent and export problems led them to eschew patents entirely. PGP 3 introduced use of the CAST-128 (a.k.a. CAST5) symmetric key algorithm, and the DSA and ElGamal asymmetric key algorithms, all of which were

unencumbered by patents.

After the Federal criminal investigation ended in 1996, Zimmermann and his team started a company to produce new versions of PGP encryption. They merged with Viacrypt (to whom Zimmermann had sold commercial rights and who had licensed RSA directly from RSADSI) which then changed its name to PGP Incorporated. The newly combined Viacrypt/PGP team started work on new versions of PGP encryption based on the PGP 3 system. Unlike PGP 2, which was an exclusively command line program, PGP 3 was designed from the start as a software library allowing users to work from a command line or inside a GUI environment. The original agreement between Viacrypt and the Zimmermann team had been that Viacrypt would have even-numbered versions and Zimmermann odd-numbered versions. Viacrypt, thus, created a new version (based on PGP 2) that they called PGP 4. To remove confusion about how it could be that PGP 3 was the successor to PGP 4, PGP 3 was renamed and released as PGP 5 in May 1997.

OpenPGP

Inside PGP Inc., there was still concern about patent issues. RSADSI was challenging the continuation of the Viacrypt RSA license to the newly merged firm. The company adopted an informal internal standard called "Unencumbered PGP": "use no algorithm with licensing difficulties". Because of PGP encryption's importance worldwide (it is thought to be the most widely chosen quality cryptographic system), many wanted to write their own software that would interoperate with PGP 5. Zimmermann became convinced that an open standard for PGP encryption was critical for them and for the cryptographic community as a whole. In July 1997, PGP Inc. proposed to the IETF that there be a standard called OpenPGP. They gave the IETF permission to use the name OpenPGP to describe this new standard as well as any program that supported the standard. The IETF accepted the proposal and started the OpenPGP Working Group.

OpenPGP is on the Internet

Standards Track and is under active development. The current specification is RFC 4880 (November 2007), the successor to RFC 2440. Many e-mail clients provide OpenPGP-compliant email



security as described in RFC 3156.

The Free Software Foundation has developed its own OpenPGP-compliant program called GNU Privacy Guard (abbreviated GnuPG or GPG). GnuPG is freely available together with all source code under the GNU General Public License (GPL), and is maintained separately from several Graphical User Interfaces (GUIs) that interact with the GnuPG library for encryption, decryption and signing functions (see KGPG, Seahorse, MacGPG). Several other vendors have also developed OpenPGP-compliant software.

Network Associates acquisition

In December 1997, PGP Inc. was

acquired by Network Associates, Inc. ("NAI"). Zimmermann and the PGP team became NAI employees. NAI was the first company to have a legal export strategy by publishing source code. Under NAI, the PGP team added disk encryption, desktop firewalls, intrusion detection, and IPsec VPNs to the PGP family. After the export regulation liberalizations of 2000 which no longer required publishing of source, NAI stopped releasing source code.

In early 2001, Zimmermann left NAI. He served as Chief Cryptographer for Hush Communications, who provide an OpenPGP-based e-mail service, Hushmail. He has also worked with Veridis and other companies. In October, 2001, NAI announced that its PGP assets were for sale and that it was suspending further development of PGP encryption. The only remaining asset kept was the PGP E-Business Server (the original PGP Commandline version). In February 2002, NAI canceled all support for PGP products, with the exception of the renamed commandline product. NAI (now McAfee) continues to sell and support the product under the name McAfee E-Business Server.

Current situation

In August 2002, several ex-PGP team members formed a new company, PGP Corporation, and bought the PGP assets (except for the command line version) from NAI. The new company was funded by Rob Theis of Doll Capital Management (DCM) and Terry Garnett of Venrock Associates. PGP Corporation supports existing PGP users and honors NAI's support contracts. Zimmermann now serves as a special advisor and consultant to PGP Corporation, as well as continuing to run his own consulting company. In 2003, PGP Corporation created a new server-based product called PGP Universal. In mid-2004, PGP Corporation shipped its own command line version called PGP Command Line, which integrates with the other PGP Encryption Platform applications. In 2005, PGP Corporation made its first acquisition—the German software company Glück & Kanja Technology AG, which is now PGP Deutschland AG. In 2010, PGP Corporation acquired Hamburg-based certificate authority TC TrustCenter and its parent company, ChosenSecurity, to form

its PGP TrustCenter division.

Since the 2002 purchase of NAI's PGP assets, PGP Corporation has offered worldwide PGP technical support from its offices in Draper, Utah; Offenbach, Germany; and Tokyo, Japan.

On April 29, 2010 Symantec Corp. announced that it would acquire PGP for \$300 million with the intent of integrating it into its Enterprise Security Group. This acquisition was finalized and announced to the public on June 7, 2010. The source code of PGP Desktop 10 is available for peer review.

PGP Corporation encryption applications

While originally used primarily for encrypting the contents of e-mail messages and attachments from a desktop client, PGP products have been diversified since 2002 into a set of encryption applications which can be managed by an optional central policy server. PGP encryption applications include e-mail and attachments, digital signatures, laptop full disk encryption, file and

folder security, protection for IM sessions, batch file transfer encryption, and protection for files and folders stored on network servers, and, more recently, encrypted and/or signed HTTP request/responses by means of a client side (Enigform) and a server side (mod openpgp) module. There is also a Wordpress plugin available, called wp-enigform-authentication, that takes advantage of the session management features of Enigform with mod_openpgp.

The PGP Desktop 9.x family includes PGP Desktop Email, PGP Whole Disk Encryption, and PGP NetShare. Additionally, a number of Desktop bundles are also available. Depending on application, the products feature desktop e-mail, digital signatures, IM security, whole disk encryption, file and folder security, self decrypting archives, and secure shredding of deleted files. Capabilities are licensed in different ways depending on features required.

The PGP Universal Server 2.x management console handles centralized deployment, security policy, policy enforcement, key

management, and reporting. It is used for automated e-mail encryption in the gateway, and manages PGP Desktop 9.x clients. In addition to its local keyserver, PGP Universal Server works with the PGP public keyserver—called the PGP Global Directory—to find recipient keys. It has the capability of delivering e-mail securely when no recipient key is found via a secure HTTPS browser session.

With PGP Desktop 9.x managed by PGP Universal Server 2.x, first released in 2005, all PGP encryption applications are based on a new proxy-based architecture. These newer versions of PGP software eliminate the use of e-mail plug-ins and insulate the user from changes to other desktop applications. All desktop and server operations are now based on security policies and operate in an automated fashion. The PGP Universal server automates the creation, management, and expiration of keys, sharing these keys among all PGP encryption applications.

The current shipping versions are PGP Desktop 10.2.0 (Windows and Mac-OS Platforms), and PGP Universal 3.2.0.

Also available are PGP Command Line, which enables command line-based encryption and signing of information for storage, transfer, and backup, as well as the PGP Support Package for BlackBerry which enables RIM BlackBerry devices to enjoy sender-to-recipient messaging encryption.

New versions of PGP applications use both OpenPGP and the S/MIME, allowing communications with any user of a NIST specified standard.

Further reading

Garfinkel, Simson (1991-12-01). *PGP: Pretty Good Privacy*. O'Reilly & Associates. ISBN 1-56592-098-8.

Zimmermann, Phil (1991-06). *Why I Wrote PGP*. (<http://www.philzimmermann.com/EN/essays/WhyIWrotePGP.html>)

MORE UBUNTU!

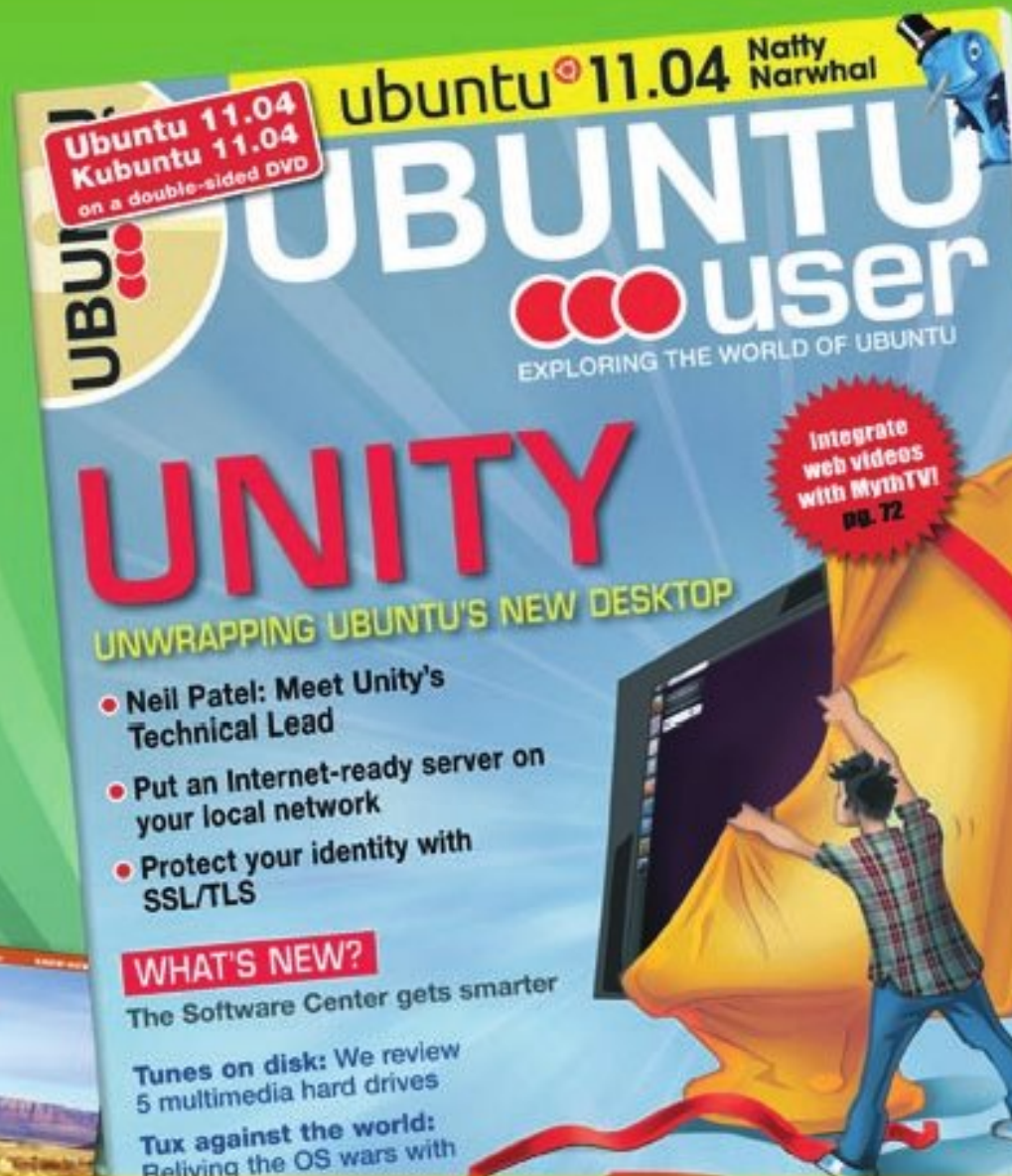
Can't get enough Ubuntu?
We've got a whole lot more!

Ubuntu User is your roadmap to the Ubuntu community. In the pages of **Ubuntu User**, you'll learn about the latest tools, best tricks, and newest developments in the Ubuntu story.

DON'T MISS ANOTHER ISSUE!



UBUNTU-USER.COM/SUBSCRIBE-NOW



FOLLOW US ON

TWITTER: UBUNTUSER

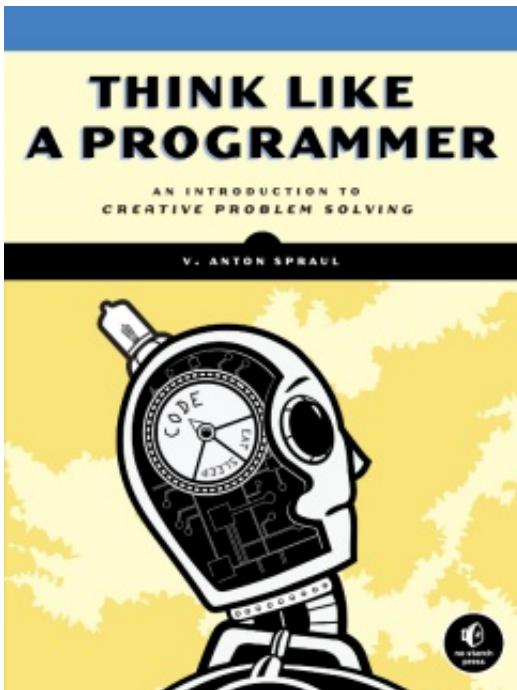
FACEBOOK: UBUNTUSERMAG



BOOK REVIEW

Written by Lucas Westermann

Think Like A Programmer



Think Like A Programmer by V. Anton Spraul (published by **No Starch Press**) is a 226 page, 8 chapter long book, of typical “technical manual” dimensions. That's essentially where the similarities to most technical instruction books end. As opposed to books like “Programming PHP” or “Beginning Python”, this book doesn't aim at teaching you a specific programming language.

Indeed, it assumes a fair amount of knowledge of C++. Which isn't to say that knowledge is necessarily required for this book. If you're comfortable piecing together a working knowledge of C++ through examples, or can convert solutions into a language you're more comfortable with, it should still prove rather enlightening. After all, the actual goal of this book is to introduce you to creative problem solving. Specifically, it tries to teach you a method of approaching complex problems, and to break it into its constituent parts, to allow you to make better progress.

Method

The first thing that struck me about this book is the method with which they teach. The first chapter is dedicated solely to the strategies for solving problems. They do this by offering puzzles and riddles which, while solvable, aren't immediately answerable by the reader. The reason for this is due to the listener assuming implications that don't exist. For

example, assuming the possible actions listed in a problem are the only actions you can take. By exposing you to these sorts of problems from the beginning, the author is hoping to make you more aware of the assumptions you're making. In doing this, you then also become more aware of your specific method for answering problems, allowing you to better follow the later problems and discussions. The answers to each puzzle are also clearly explained in the author's own methodology, while still encouraging the reader to explore other possible solutions. As such, I find this is one of the most helpful books I've read, due to the fact that it guides you towards designing a system for yourself, as opposed to encouraging a mindset where there can be only one correct method.

Exercises

Besides discussing the strategies behind problem solving, this book also offers a slew of actual exercises to solve using C++.

Obviously, almost all of these problems can all be solved in most programming languages; the answers supplied are simply in C++. Each successive chapter is dedicated to driving the original strategy home. This means that it helps you to break down a problem, to notice which sections of a problem you may have already answered, and how to keep up the motivation to solve a problem. However, each chapter focuses on a specific type of problem. For example, Chapter 2 is focused on input, output and tracking the state of a program, while Chapter 6 focuses on recursion. This approach to teaching helps cement the basic strategies for solving the problems, while helping you apply them to a multitude of practical examples. It also helps you understand certain aspects of programming (especially in C++) that can sometimes be abstract, or confusing to comprehend.

Diagrams

One aspect of the book that I found made it much more

approachable is the liberal use of diagrams, tables, and thought processes. For example, for some of the more convoluted puzzles, the book usually offers a diagram to help explain what is meant. This can help you to understand, but also shows you how to translate word-based problems into visual representations. As such, if you're a big fan of visualization, or simply a visual learner, the method and explanations in this book can be invaluable.

Explanations

For those who are most comfortable with written explanations, then you will also not be disappointed. The author has a writing style that is both easily understandable, and enjoyable to read. As opposed to most reference books, the author imbues a certain level of life to his prose, allowing otherwise dry and technical explanations to be entertaining.

Conclusion

Overall, I find this book is an excellent addition to any library of reference books, so long as you

sincerely wish to learn. If you approach this book with the intent to learn as much about creative approaches to solving problems, you'll probably find that working through this book goes much quicker (and is much more entertaining) than you might have first assumed. If, however, you're already comfortable solving complicated problems, or have little interest in developing an approach to solving those sorts of problems, this book may be of little interest to you. For anyone uncertain of which group they belong to, I offer this advice: find a physical copy of the book that you can flip through, and looking through Chapters 1 & 2 should give you a very good idea of what to expect. If you have any specific questions, you're welcome to email me at lswest34@gmail.com, and I will do my best to answer them.



Lucas has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. You can email Lucas at: lswest34@gmail.com.

the finest in geek entertainment

Catalog

Catalog

- Art, Photography, Design
- Business
- For Kids
- General Computing
- Hardware and DIY
- LEGO®
- Linux, BSD, Unix
- Mac
- Manga
- Programming
- Science & Math
- Security
- System Administration

Free ebook edition with print book purchase from nostarch.com!

Shopping cart

View your shopping cart.

User login

- Log in
- Create account

Bestsellers

New!

Beneath OS X's glossy surface, the fearless user will find lots of powerful features that aren't immediately obvious. **Master Your Mac** aims to help users cross that divide.

The real fun of LEGO building lies in creating your own models—from choosing the subject to clicking that final brick into place. Learn how in **The Unofficial LEGO Builder's Guide**.

Whether you're brand new to LEGO or have been building for years, unleash your imagination with **The LEGO Adventure Book!** Learn to build robots, trains, medieval villages, and much more.

The Unofficial LEGO Technic Builder's Guide is filled with building techniques and tips for creating strong yet elegant machines and mechanisms.

From the brilliant mind of Japanese artist Buntpei Yonfuji comes **Wonderful Life with the Elements**, an illustrated guide to the periodic table that gives chemistry a friendly face.

In **Super Scratch Programming Adventure!**, kids learn programming fundamentals as they make their very own playable video games. They'll create projects

Coming Soon [\(see all\)](#)

Full of fun examples and color illustrations, **Python for Kids** is a playful introduction to Python that will help any beginner get started with programming.

Learn You Some Erlang for Great Good! is a hilariously illustrated guide to the concurrent functional programming language. As you laugh along with Hebert's brilliantly quirky drawings, you'll effortlessly pick up this complex language and have fun while you're at it.

Whether you're just getting started with GIMP or working to master GIMP's more complex features, you'll find the answers you're looking for in **The Book of GIMP**.

Arduino Workshop takes you through 60 electronics projects that show the full range of cool stuff you can do with Arduino. Each project builds upon the last, increasing in difficulty as you gain more knowledge and experience.

By combining comics with real-world examples, **The Manga Guide to Regression Analysis** teaches you effective ways to analyze data and make predictions.

nostarch.com

full circle magazine #67 56

[contents](#) ^



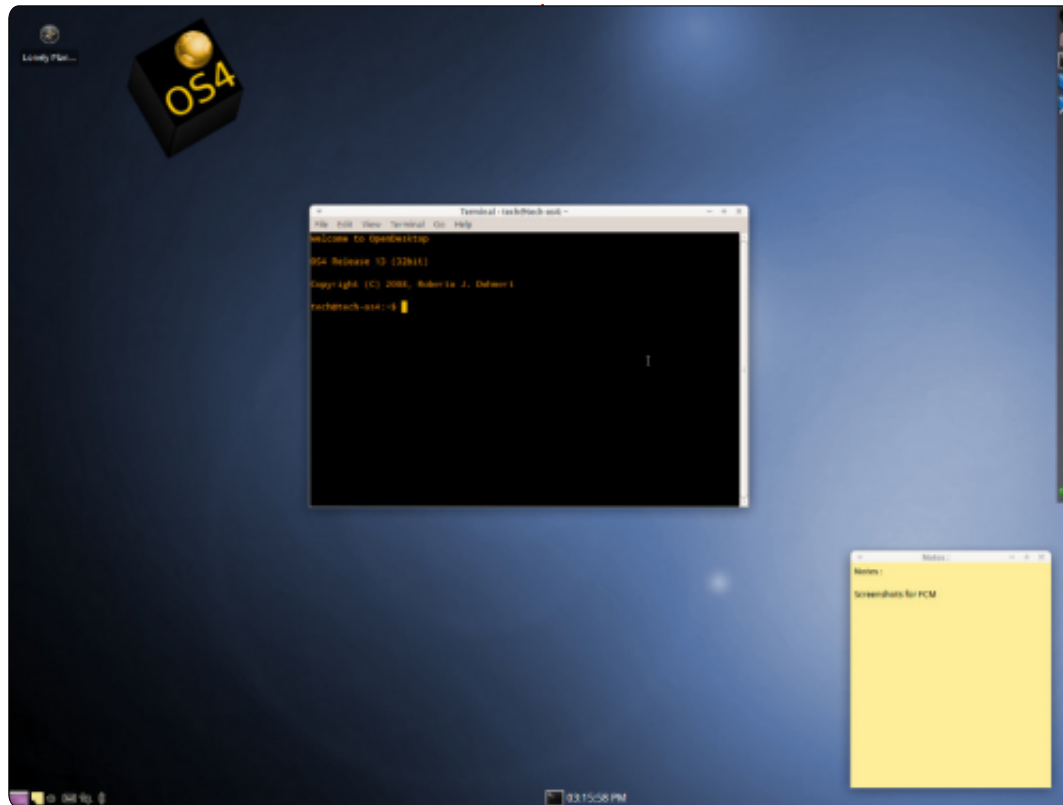
REVIEW

Written by Rich Dennis

OS4 OpenWorkStation

XFCE has grown in popularity as a desktop environment since the release of the GNOME 3.x series and the controversial changes it has made to a well-known and heavily used DE. Debian is rumored to be considering XFCE as the default desktop for the upcoming 7.x release Wheezy. Respins (Re-Spins) and remasters featuring XFCE are becoming increasingly common, due to XFCE's intuitive interface, stability, flexibility, and ability to conform to a wide variety of contemporary and legacy hardware platforms.

Roberto Dohnert, Brian Johnson and Sebastian Wells created the first version of what would evolve through several iterations into OS4 OpenDesktop in 2005—it was called OS4u and based on Gentoo Linux, backed by a commercial developer. Fast-forwarding to 2008, after the dissolution of that partnership, Dohnert and PC / OpenSystems LLC created the first version of their take on the Linux desktop,



PCOS 8.04 in 2008. Moving from Gentoo to an Ubuntu base offered the vast array of Debian apps and advanced toolset as well as Ubuntu's well-known ease-of-use and growing popularity. Ubuntu was innovative and well-regarded. Fast-forward again to 2012: Dohnert will be releasing OS4 OpenWorkstation 13 Update 1 on December 1 2012, which will be

based on Xubuntu 12.04 and the Linux 3.5.x series kernel. In these four years, the name may have changed, but the important elements have stayed the same for this distro: an adherence to XFCE, ease of use, great hardware compatibility, a simplified but innovative interface, and an excellent set of default applications.

Most recent Canonical derivatives and respins have a similar feel and generic look; Dohnert's vision for the desktop has always been unique and idiosyncratic. His first choices for a GUI were variations on interfaces popular at the time: NeXTStep, BeOS, AmigaOS and SGI Irix. Although the first versions of PCOS in 2008 had a BeOS-style layout and theme, OS4 has moved towards NextSTEP in its appearance and overall theme. In an interview published on the distro's website—part of which was published online on Distrowatch.com—Dohnert mentioned his history with the venerable OSX precursor and his hopes to leverage the stability of the Linux kernel to bring “ease of use to the desktop.” And a recent problem with some members of the Amiga user community has been resolved; OS4 13 OpenDesktop includes integrated modules for mounting AmiFS media, and the team has assisted AmigaOS in improving its ext4 driver.





03:19:38 PM

According to the newly created installation guide, OS4 OpenDesktop runs best on machines of at least these specifications, and better on the recommended hardware :

32 bit system requirements / recommendations:

Processor PIII 800 MHz / Intel Core Duo or higher
RAM 512 MB / 1 GB +
Hard Drive 10 GB / 15 GB +
Standard VGA adapter / 3D Graphics Card

64 bit system requirements / recommendations:

AMD64 or Intel 64 1 GHz Dual Core
RAM 1 GB + / 2 GB +
Hard Drive 10 GB + / 15 GB +
Standard VGA adapter / 3D Graphics Card

The ISOs for the OS4 OpenDesktop 13 32-bit and 64-bit editions weigh in at 1.3 and 1.4 GB respectively. Once burned to a flash drive or DVD, the OS4 Live

session should prove familiar to an experienced Ubuntu user, and intuitive to the newbie – booting quickly to a serene blue desktop decorated with the OS4 logo in the upper left hand corner. Dohnert calls this environment the OpenDesktop Workspace Manager. While the usual XFCE icon sets and themes are available, two custom themes, Graphite (the default) and Sky, are included, and a custom icon set, OS4 Visual Subsystem. The bottom panel, with a left-hand notification area and window menu, workspace switcher (2 workspaces by default) in the middle, and Orage clock on the right is called the applet bar. The vertical, opaque panel on the upper right-hand side is called the Shelf: it features a Trashcan on the bottom, Menu with icons that can be dragged to the panel, and File Manager launcher on the top.

The system, based on Xubuntu 12.04, is generally quick, responsive, and very stable. Thunar

is the default file manager and handles the desktop: the XFCE Task Manager shows typical resource consumption (like its parent, OS4 is very lean, averaging about 11% processor usage on single-core Pentium 4 and 15% memory useage on 2.5 GB of RAM while writing this article on Libreoffice Writer with five tabs open in Google Chrome), and, at the epicenter, the Settings Manager for under-the-hood changes. The application selection is typical, with a few interesting wrinkles: Nokia Maps is included, and the Nixnote Evernote client (a personal favorite). OS4 includes the Illumination Software Creator, designed by Brian Lunduke for the novice programmer, and Eclipse and Netbeans for the more advanced coder. But the typical array of productivity software is included out-of-the-box, and the average user will find him-or-herself able to get to work almost immediately with an enhanced suite of Office-compatible and

Web-capable apps: Google Chrome, the Evolution email and groupware client, Abiword, Gnumeric, Orage. The inclusion of Remastersys, to allow the creation of customized OS4 configurations and full system backups, is another welcome inclusion.

There are some differences between OS4 and a typical XFCE desktop : the top-level menu is absent—right-clicking on the desktop brings up the Applications menu. Additionally, although there's a minimized-window menu in the Applet bar, open applications also iconize to the desktop. Right-clicking on iconized apps brings up the usual context menu, but these can't be moved to another workspace or closed. Dohnert says that this is an upstream XFCE bug that will be addressed in Update 1.

After examining the Live session, and moving forward to

installing the system, the steps are, again, familiar and intuitive: selecting a timezone, configuring the keyboard, partitioning the disk(s), creating a user account. After these steps are completed, the new OS4 user is ready to get started browsing the Internet, enjoying their music libraries with gtkpod and Banshee, tweet with Hotot, and get organized with Evolution.

The company is progressing. It has a wide user base and a new four-man development team—Dohnert, Cristobal Molina, Seth Forstal and Mike Vail. They offer OS4 OpenDesktop as a free download, but also provide paid support for customers through the OS4 Store. They have also diversified from XFCE, offering a KDE version for European users who demanded that particular desktop environment and suite of QT-based applications. OS4 OpenServer is available also for the small-business and home server market, and soon-to-come, Trusted OS4 1.0, featuring the NSA's Linux initiatives, DoD level drive wiping and encryption software, and access card controls. The team and their OS4 storefront are responsive to their user base,

offering custom installs for particular hardware configurations, and, for the purposes of this article, Dohnert himself was very helpful and accessible. The main version will not be ignored of course: plans to increase compatibility with Apple hardware and with recent updates to the AmigaOS, should make the future of this distro bright.



Rich Dennis is a network engineer, aspiring writer and burgeoning Linux enthusiast who prefers XFCE as a desktop environment generally and Xubuntu 12.04 specifically.



QUICK REVIEW

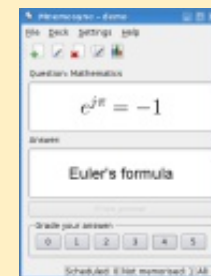
by Peter Liwyj

It isn't often that a program totally supprises me. **Mnemosyne** really did. I thought it was a basic flash card program but the way it presents the cards actually forces you to learn and retain information. Cards that you do know well are pushed to the background and cards you do not know will show up more often. Mnemosyne will not waste your time on things you know but will check and make sure you haven't forgotten anything. Probably the most useful feature is that you can flip cards to make the questions become answers, and answers become questions, effectively doubling the card set and making you learn things backwards and forwards.

Mnemosyne is a flash card program that may appear very basic when you first look at it, but it actually has some very useful features. It supports different scripts so you can use it to learn languages with different alphabets, three sided cards are supported, you can use LaTeX to write out math formulas and, if you need it, pictures, sounds and HTML can be attached. It can also run off a usb stick and can cross platforms including Android and Blackberry devices, you don't have to waste your bus commute to school anymore.

There is a small community who have made card sets and posted them on the Mnemosyne site for free download. Personally I have found the act of making the cards helps you to better learn the information. So if you find yourself struggling with Latin and medical terminology, or need to learn the hot key shortcut commands for Blender, give Mnemosyne a try. I think you will quickly learn to respect this well hidden and little known program with the strange name.

<http://www.mnemosyne-proj.org>





LETTERS

If you would like to submit a letter for publication, compliment or complaint, please email it to: letters@fullcirclemagazine.org. PLEASE NOTE: some letters may be edited for space.



Join us on:



goo.gl/FRTMI



facebook.com/fullcirclemagazine



twitter.com/#!/fullcirclemag



linkedin.com/company/full-circle-magazine



ubuntuforums.org/forumdisplay.php?f=270

Actually...

I enjoyed the Ask The New Guy column for this month (on updating packages), however there is one minor error, and one other thing I'd like to point out. In the article, when Copil is talking about installing a program through the Software Center, he says "Choose the software you want from the results, and click INSTALL. Once you give it your root password, the program installs." Actually it's your password (as you don't use the root account for anything)--not the root password.

The other thing I'd like to point out is that you can use sudo apt-get dist-upgrade (the difference between dist-upgrade and upgrade is that dist- will find and install any new dependencies automatically).

Patrick

Nexus 7

I'm a big fan of the Nexus 7 - since getting one a few months ago, it's become my most used computer. But there's one big thing missing from the otherwise well-designed hardware: video-out.

My Galaxy Nexus phone, for instance, supports video output through the USB port (after buying a pretty inexpensive USB/HDMI adaptor). That doesn't work with the Nexus 7, however.

The result - there's no way that I'm aware of to view stored videos on a Hi-Def TV. In a more business-like (or education) context, there's no way to display presentations or other material on a digital projector. And that makes it MUCH less useful in those business or education contexts where it would otherwise be a natural.

Alan Zisman

SoftMaker 2008 Office

Just to let everybody know, SoftMaker of Germany is giving away their SoftMaker 2008 office suite which includes word processing, spreadsheet, presentation, and VBA coding functions.

Normally this suite costs \$100 to \$200, but the lesser version can be found at <http://www.softmaker.de/lh-down-en.htm>. Be aware that the site asks for name and email address, at which time a serial number and download link is emailed. You can expect at least one email per week pitching an upgrade, but you can opt out after receiving the freebie.

Also, although it appears a 386 .deb file is the only offering for Linux, my testing showed it works just fine on 64-bit versions, too. In work I've done with it, the suite opens quicker than LibreOffice, and performs admirably.

Art Schreckengost

Magic Numbers

I've really been enjoying the "Closing Windows" series, and was particularly excited when I saw the topic of File Associations (FCM#66). After reading it, I had the impression that Linux works just like Windows. This disappointed me, because I thought that internally there was a better system in Linux. I remember seeing something about "Magic Numbers" that are supposed to play some role in opening files, perhaps even getting rid of the need for extensions. I'd love to see something in FCM discussing this.

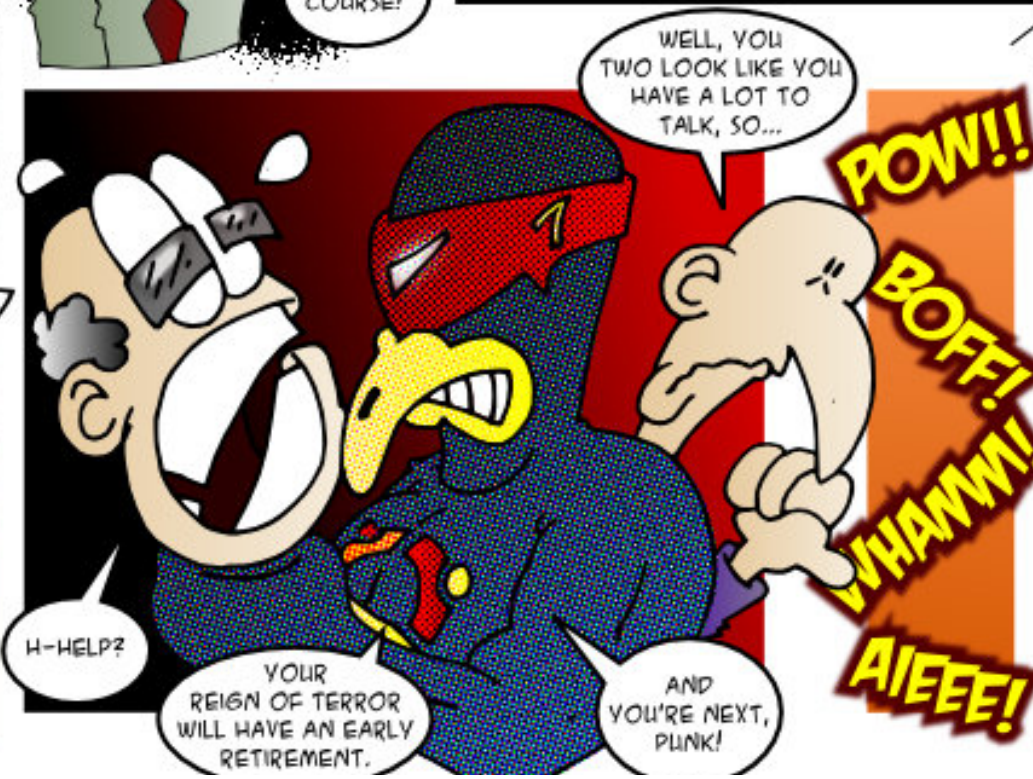
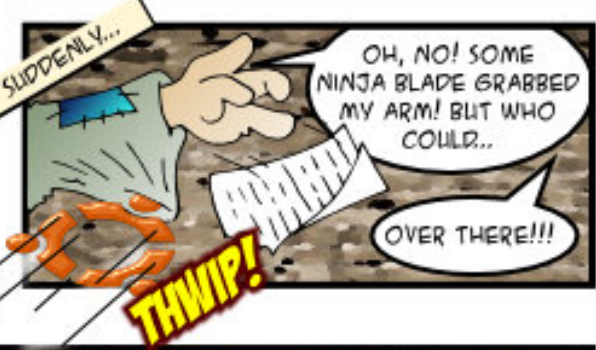
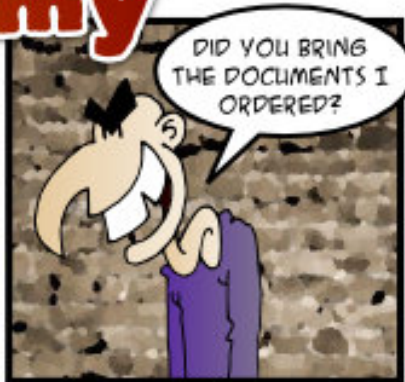
Ernie DeVries



Tuxidermy

PENSADOR LOUCO

THE WORLD WAS RUINED BY THE GREED OF MANKIND, BUT SOME CRIMINALS JUST COULDN'T STOP



YOU THOUGHT RIGHT. IT'S MONOPOLY GAME MONEY.



Q&A

Compiled by Gord Campbell

If you have Ubuntu-related questions, email them to: questions@fullcirclemagazine.org, and Gord will answer them in a future issue. Please include as much information as you can about your problem.

Q After installing Ubuntu 12.10, then the Nvidia Current "Additional Driver", I have a shrunken screen with no panel and no launcher.

A (Thanks to **RaduStoica** on Launchpad) Install linux-headers-generic before installing nvidia-current. Gord adds: *the failure to include linux-headers-generic during installation is extremely puzzling.*

Q If I wanted to start ubuntu 12.10 from a pen drive, what would be the best size to buy?

A If you are going to use just the pen drive to install Ubuntu, 1 GB is enough. If you are going to run Ubuntu from the pen drive, 4 GB or 8 GB are good. You are limited to 4 GB of "persistent" space, so part of an 8 GB drive will be wasted, but the cost difference is minor.

Q When I try to start Shotwell, nothing happens.

A Try opening a terminal, and enter the command:

```
shotwell
```

If there is an error, it should appear. (And it did: "missing library file". Reinstalling the library file got Shotwell running.)

Q How can I stop my computer from suspending or hibernating while I am running my backup program?

A (Thanks to **Paddy Landau**) Install Caffeine (<https://launchpad.net/~caffeine-developers/+archive/ppa>), which turns off sleep mode while programs that you specify are running.

Q Issue 64 of Full Circle Magazine had information about the Wirenet trojan. What should we do about it?

A See <http://askubuntu.com/questions/181930/what-to-do-regarding-backdoor-wirenet-1>

In short, if you don't have a WIFIADAPT folder, you don't have the trojan. If you create a read-only *file* with that name (in Home), the trojan can not be installed. (Note that the name is all upper-case.)

Q Can I install Microsoft Office 2007 using PlayOnLinux, and have it automatically open for .DOC (and similar) files?

A Yes, see this tutorial: <http://ubuntuforums.org/showthread.php?t=1940522>

Q My computer has multiple hard drives in a RAID setup. Do I still need to do backups?

A Yes, you do! People will delete files and then decide they need them, and RAID provides no protection against this.

Q Just installed Xubuntu 12.10 on my laptop. Is there a way to make the battery applet show the battery percentage or/and the time?

A (Thanks to **badhorse** in the Ubuntu Forums) Install xfce4-battery-plugin and add the new battery item to your panel.

Q Sometimes VLC plays video fine, but there is no audio.

A Under Audio, there is a "mute" setting -- which does not show if it is enabled! Try clicking on it.

Q I've downloaded LibreOffice to install on an older release of Ubuntu that had OpenOffice. I've removed OO. What directory should I extract LibreOffice to?

A

Don't extract it, open it with gdebi. (Right-click on the downloaded file.)

Q My graphics card is an nVidia GeForce4 TI4200 with AGP8. I want to have dual mode with my monitor at 1024x768 and the TV at 800x600. I can't find an Nvidia driver for my card which works with the current Ubuntu.

A Your video card was first produced more than 10 years ago. My rule of thumb is that PC technology is good for seven years.

(I got a new computer in 1990, then a new computer in 1997, and again in 2004. That one was passed to my wife, and she got a new computer in 2011.)

Q Where can I find some free/cheap Linux stickers? I want to put tux on my netbook.

A (Thanks to **QuickSphinx** in the Ubuntu Forums) I ended up buying 3 here for 10 bucks:

http://tuxstickers.ptaff.ca/step1/?lang=en_CA

It really depends on your location. Google can help. Kudos to System 76.

Q What's the secret for playing iTunes podcasts in Ubuntu?

A To the best of my knowledge, Ubuntu users have no access to the iTunes Music Store. If you want podcasts which are available only through iTunes, you really need to run Windows or an

Apple OS.

I use Miro to collect and play podcasts, including podcasts which are available as RSS subscriptions from web sites such as Revision3. There are other options in the Software Centre; Miro certainly is not perfect.

Tips and Techniques



Distro Testing

On average, I have a look at one distro or version of Linux a month. Over time, I've developed a checklist of things to try on my desktop and laptop.

Video: Can I see a graphical interface? (No is a deal breaker!) Is the monitor resolution correct? Can I install a proprietary driver?

Wireless: Does my wireless adapter work?

Audio: Can I hear sound from my earphones and, on my laptop, the speakers? Can I record audio using Audacity?

Media: Can I play MP3s and
FLACs? Can I watch Youtube
videos? Can I watch DVDs?

Firefox: Can I install extensions to auto-select my choice of resolution for Youtube videos, and to download Youtube videos?

Sharing: Can I access shared folders on other computers on my network? Can I share a local folder for other computers to access?

Printing: Can I easily use my networked Brother laser printer?

Webcam: Does Cheese work with my webcam?

Camera: Can I transfer pictures from my elderly Canon DSLR and videos from my video camera?

Conky: Can I display the temperatures of various components, and other Conky goodness?

Applets: Can I see the weather for my location? Can I access my Dropbox folders?

Applications: Can I install Google Chrome, the Adobe Reader and Cinelerra from their respective

sources?

Stability: Are there any crashes or other anomalies?

Shutdown: Do shutdown and restart work as expected?

So now you're asking, how does Ubuntu 12.10 measure up? I'll look at only the problem areas.

I installed a proprietary video driver (nvidia current) on my desktop computer, but it didn't produce a working system. After rebooting, I decided the fastest solution was to reinstall and avoid the proprietary driver. For 99% of the populace, this would be a deal breaker, and the end of their experiment with Linux. (See the Question above on how this was resolved.) My laptop, with ATI graphics, did not offer a proprietary driver. (Oddly, the "additional drivers" in Software Centre claimed my wireless adapter was not working, but Software Centre was connected using that wireless adapter!)

When I ran my standard Conky script, the system would lock up hard in anywhere from 30 minutes to two hours. I'm still trying to

track down the problem. It's not just Conky, the system locks up when I have not run Conky. I had one session of more than 20 hours before I did a normal reboot, but I have not been able to repeat it.

For DVD playback, I had to, as expected, follow the instructions found here:
<https://help.ubuntu.com/community/RestrictedFormats>

I downloaded the Adobe Reader, and tried to install it using Software Centre. An error message flashed by much too quickly to read, then Software Centre offered to install the application (again) -- with the same result. The Dash offered to run Adobe Reader, but it did not run. When I double-clicked on a PDF, nothing happened. I could right-click on a PDF and select "Document Viewer," and that worked.

I had one crash of Compiz, which required a restart.

All in all, not too terrible. If that sounds like faint praise, it is. Sorry, I just want to get things done, not spend time debugging problems on my system. One area where I observed an improvement was in

transferring pictures from my digital camera; in previous versions, it has been hit and miss. I also liked having it make the wireless connection while I typed in my password.

Here's a pleasant note to end on. When I built my desktop system, more than three years ago, I selected a D-Link DWL-G510 PCI wireless adapter. The result is that I have never needed to connect an Ethernet cable to the computer, the wireless has worked with every distro or version of Linux I have tried.

Do you have suggestions to add to my checklist? Email suggestions/questions to:
questions@fullcirclemagazine.org.



After a long career in the computer industry, including a stint as editor of Computing Canada and Computer Dealer News, **Gord** is now more-or-less retired.



MY DESKTOP

Your chance to show the world your desktop or PC. Email your screenshots and photos to: misc@fullcirclemagazine.org and include a brief paragraph about your desktop, your PC's specs and any other interesting tidbits about your



I'm a beginner to Ubuntu having first heard of Linux a year ago. My desktop uses Docks and Conky. Very simple.

OS: Ubuntu 12.04 LTS x64

Processor: Intel Core i5 -2410 CPU, 2.3 GHz

RAM: 6GiB

Graphics: Intel HD graphics 3000

GTK+ Theme: Zukitwo

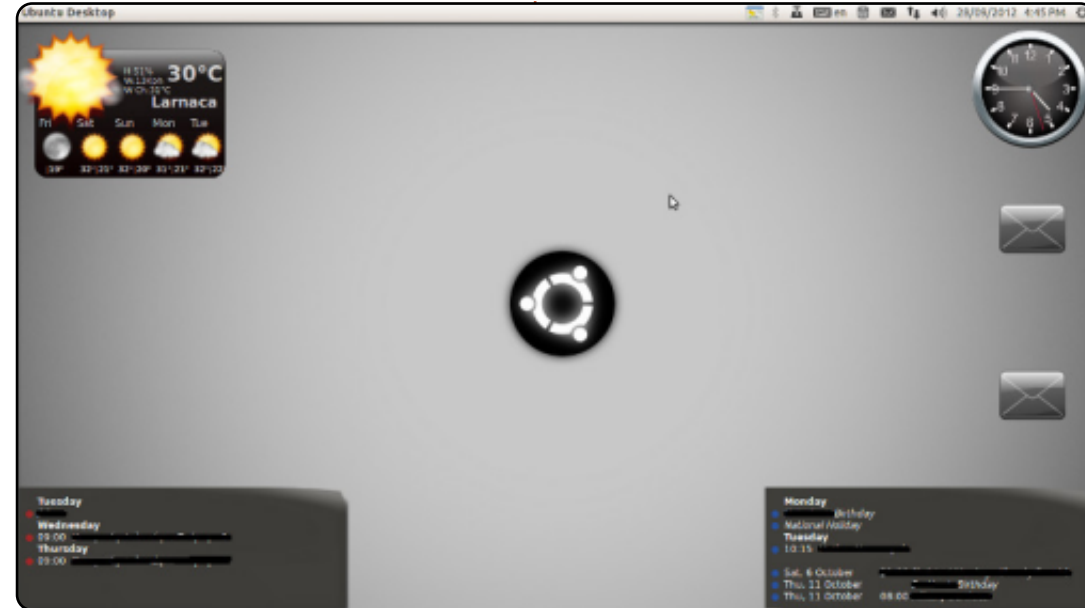
Window Theme: Zukitwo

Shell Theme: Zukitwo

Icon theme: faenza

Cursor: Shere Khan X

Cherry



This desktop configuration runs on my laptop, home PC, and office PC. As you can see, I love screenlets. I use two Mail Screenlets and two Calendar Screenlets: one is for personal use and the other for work.

Home PC (will soon be transformed into home server):

Intel Core 2 Quad Q9300 2.5 GHz

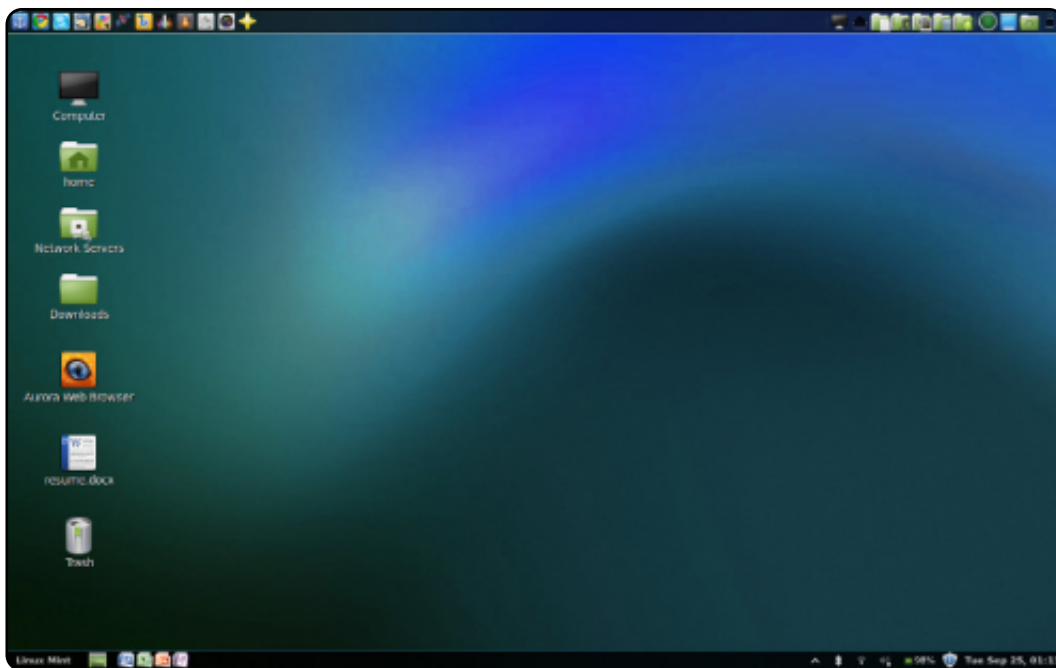
8 GB DD2 RAM

2x120 GB, 2x250 GB, 1x1.5 TB HDDs

ASUS P5K Premium Motherboard

ATI Radeon 5770 Graphics

AnGeLoS



The top panel is Dobby and the Cinnamon theme is ICS. The wallpaper should look familiar; it is actually the default wallpaper for Ubuntu 12.10, except that the hue was changed in Gimp. I use Microsoft Office only for compatibility on other computers, but I prefer LibreOffice. I use this laptop as my main computer—some light gaming like Sim Tower and Micropolis, as well as writing and designing for school.

Operating System: 32 bit Linux Mint 13 Maya Cinnamon

Icons: Mint-X (Default)

Gtk+ Theme: Adwaita

Cinnamon Theme: ICS

CPU: Intel Core 2 Duo 1.6 GHz

RAM: 2 GB

HDD: 200 GB (And shrinking)

Computer Model: Fujitsu Lifebook A-Series

Keegan



Been using Linux since 2009 and always exploring what comes out to not tie myself to just one spot.

Desktop: Moomex GTK2.x theme with Faenza-Darkest icons along with Dobby. I adjusted the transparency—using Compiz for the panel and Ubuntu tweak for everything else.

System Specs: Acer Aspire 5336 packed with Intel Celeron 900 @ 2.20Ghz, 3GB of DDR3, 250GB HDD, and 64MB of Intel GMA4500M graphics (fancy huh?)

OS: Ubuntu 10.04.2 LTS (Long Term Support) 64-bit "Lucid lynx"

System runs very quick, and is 100% reliable with Linux. My only issue is the backlight bug for these older Intel video chipsets on the laptops, which is why I cannot upgrade. However, 10.04 hasn't failed me yet, so I will stick with it until the machine dies. It's keeping me a very happy camper.

Donald McCulloch

1	4	5	6	8	B	3	C	E	9	D	A	F	0	2	7
3	F	8	0	6	1	A	9	2	7	B	4	5	C	D	E
A	E	C	2	F	0	D	7	3	8	1	5	9	6	B	4
7	B	D	9	5	E	2	4	0	C	6	F	8	A	1	3
9	1	4	C	7	3	B	A	8	E	0	6	2	D	5	F
0	A	E	5	1	C	4	8	7	F	2	D	3	B	6	9
D	6	2	F	E	5	9	0	A	4	3	B	C	7	8	1
B	3	7	8	2	D	6	F	5	1	9	C	E	4	A	0
5	9	3	E	A	6	8	1	F	D	7	0	B	2	4	C
F	7	A	D	3	4	C	2	6	B	8	9	0	1	E	5
2	8	0	4	9	F	5	B	C	A	E	1	D	3	7	6
6	C	1	B	D	7	0	E	4	3	5	2	A	9	F	8
8	2	B	3	C	A	F	5	1	0	4	7	6	E	9	D
C	5	6	7	4	8	E	D	9	2	A	3	1	F	0	B
E	0	9	1	B	2	7	3	D	6	F	8	4	5	C	A
4	D	F	A	0	9	1	6	B	5	C	E	7	8	3	2

SUDOKU

CODE WORD

R	E	N	A	M	I	N	G			M	U	T	E
	X		R		N		O		A		O		
S	P	A	S	M	S			R	E	T	Y	P	E
	L		O		E	V	E		R		A		
S	O	W	N		R			S	N	O	O	Z	E
	D				T				N				
B	E	F	I	T	S			H	U	S	H	E	D
			N					E				J	
S	H	Y	E	S	T			P		P	R	E	Y
	A		X		R	U	T		I		C		
C	I	C	A	D	A			A	D	O	P	T	S
	K		C		M			N		U		O	
Q	U	I	T			P	R	E	S	S	U	R	E
1	2	3	4	5	6	7	8	9	10	11	12	13	
S	O	Z	D	P	W	L	M	Y	I	T	N	J	
14	15	16	17	18	19	20	21	22	23	24	25	26	
U	R	H	C	K	Q	G	E	B	X	V	F	A	
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z													



Codeword and 16x16 Suduko puzzles are copyright, and kindly provided by, **The Puzzle Club** - www.thepuzzleclub.com



HOW TO CONTRIBUTE

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your opinions, desktops, stories, how-to's, reviews, and anything else you want to tell your fellow *buntu users. Send your articles to: articles@fullcirclemagazine.org

We are always looking for new articles to include in Full Circle. For help and advice please see the **Official Full Circle Style Guide**: <http://url.fullcirclemagazine.org/75d471>

Send your **comments** or Linux experiences to: letters@fullcirclemagazine.org
Hardware/software **reviews** should be sent to: reviews@fullcirclemagazine.org
Questions for Q&A should go to: questions@fullcirclemagazine.org
Desktop screens should be emailed to: misc@fullcirclemagazine.org
... or you can visit our **forum** via: fullcirclemagazine.org



FCM#68

Deadline:
Sunday 09th Dec. 2012.
Release:
Friday 28th Dec. 2012.



Full Circle Team



Editor - Ronnie Tucker
ronnie@fullcirclemagazine.org
Webmaster - Rob Kerfia
admin@fullcirclemagazine.org
Podcast - Les Pounder & Co.
podcast@fullcirclemagazine.org

Editing & Proofreading
Mike Kennedy, Lucas Westermann,
Gord Campbell, Robert Orsino,
Josh Hertel, Bert Jerred

Our thanks go to Canonical, the many translation teams around the world and **Thorsten Wilms** for the FCM logo.

Getting Full Circle Magazine:



EPUB Format - Recent editions of Full Circle have a link to the epub file on the downloads page. If you have any problems with the epub file, you can drop an email to: mobile@fullcirclemagazine.org



Google Currents - Install the Google Currents app on your Android/Apple devices, search for 'full circle' (within the app) and you'll be able to add issues 55+. Or, you can click the links on the FCM download pages.



Ubuntu Software Centre - You can get FCM via the Ubuntu Software Centre: <https://apps.ubuntu.com/cat/>. Search for 'full circle', choose an issue, and click the download button.



Issuu - You can read Full Circle online via Issuu: <http://issuu.com/fullcirclemagazine>. Please share and rate FCM as it helps to spread the word about FCM and Ubuntu Linux.



Ubuntu One - You can now have an issue delivered to your free Ubuntu One space by clicking the 'Send to Ubuntu One' button which is available on issues 51+.